KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering

Development of a Core Management Tool for the MYRRHA Irradiation Research Facility

David Jaluvka

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor in Engineering

February 2015

Development of a Core Management Tool for the MYRRHA Irradiation Research Facility

David JALUVKA

Examination Committee:
Prof. dr. ir. J. Vandewalle, chairman
Prof. dr. ir. S. Vandewalle, supervisor
Prof. dr. ir. W. D'haeseleer, supervisor
Dr. ir. G. Van den Eynde, co-supervisor
(SCK•CEN, Belgium)
Prof. dr. P. De Causmaecker
Prof. dr. ir. D. Roose
Prof. dr. ir. G. Vanden Berghe
Prof. dr. ir. E. Mund
(Université Libre de Bruxelles, Belgium)
Prof. P. Ravetto
(Politecnico di Torino, Italy)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor in Engineering

February 2015

© 2015 KU Leuven – Faculty of Engineering
Uitgegeven in eigen beheer, David Jaluvka, Celestijnenlaan 200A box 2402, B-3001 Heverlee (Belgium)

Alle rechten voorbehouden. Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotokopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaande schriftelijke toestemming van de uitgever.

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm, electronic or any other means without written permission from the publisher.

ISBN 978-94-6018-971-5

D/2015/7515/27

Preface

This text was written as a part of my PhD at the Department of Computer Science at the University of Leuven (KU Leuven) in Belgium. The work was carried out between October 2009 and October 2014 in the Nuclear Systems Physics (NSP) expert group at the Belgian Nuclear Research Centre (SCK•CEN) in Mol, Belgium, where the first proposal for the doctoral dissertation was drafted early in 2009.

The topic of the PhD project reflected contemporary research interests in the field of the in-core fuel management of MYRRHA, a new type of a research reactor under development at SCK•CEN. The submitted work summarizes the results of the long-time efforts made in achieving the project goals, that is, in developing a software framework for rapid evaluation and optimization of MYRRHA fuel loadings. As such, it supplements, revises, and expands the content of the following research papers, which have been published over the course of the project: Jaluvka et al., 2012, 2013a,b.

This thesis was written having two kinds of readers in mind: readers with a background in nuclear engineering but with no or very limited knowledge of optimization and readers oriented to a certain extent in the field of optimization but without any knowledge of nuclear engineering. In order to make this text understandable for both groups of audience, I tried to include in it also some basic information from both fields that may be considered too elementary for one group but rewarding for the other group. For instance, this may be the case of the reactor-physics equations listed in the second chapter. While more introductory information is often included in the text to make it more coherent and understandable for the readers as a whole, a closer description of some of the more specialized problems may be omitted here for the sake of brevity. The reader is always referred to specialized literature in such cases.

The realization of this work has been a long and laborious process that would be impossible to accomplish without help of many people. First and foremost,

I would like to thank my SCK•CEN mentor Gert Van den Eynde and my KU Leuven supervisor Professor Stefan Vandewalle for their support, wise advice, and kindness throughout the years. Without their infinite patience and generosity, I would have never gotten so far. Thank you Gert and Stefan.

My sincere thank you goes also to the other members of my supervisory committee at KU Leuven, namely, to Professor William D'haeseleer for his pertinent remarks, refreshing candor, and honest opinions, to Professor Patrick De Causmaecker for his genuine interest in the topic and helpful suggestions, and Professor Dirk Roose also for his interest. Thank you professors for your precious time spent with me discussing the project at many progress meetings I have organized during my PhD.

I thank all the members of my examination committee for carefully reading my text and for their suggestions and corrections, which have surely improved the content of this dissertation. I thank Professor Joos Vandewalle for chairing the committee.

I owe my deep gratitude to Professor Hamid Aït Abderrahim, the director of the MYRRHA project, who read my e-mail in November 2008 and thereby graciously provided me with the opportunity to pursue my PhD in Belgium. Professor Aït Abderrahim has been the prime mover of the MYRRHA project and his enthusiasm and endeavor made it possible to create a stimulating research environment and attract many professionals from all over the world to Mol, many of which have become my friends. I thank him for that also.

Very special thanks go to all my NSP coworkers which helped me in all regards along the way. A huge thank to Simon Vanmaecker for his nice personality and for his countless advice in coding in general and in Python in particular. His positive energy, impressive productivity, and discipline have always been a great motivation for me. I appreciate very much the collaboration with the neutronics specialists Alexey Stankovskiy, Edouard Mbala Malambu, and Massimo Sarotto (ENEA), who have shared their expertise with me and supported me with data. Thanks to Francesco Belloni and Diego Castelliti for answering my questions on thermal-hydraulics, to Guy Scheveneels for our discussions on nuclear safety, and to Radu Popescu for his assistance in solving computer issues. Thanks to them and Alberto Ottonello, Dario Bisogni, Wim Uyttenhove, Bart Sjenitzer, Tewfik Hamidouche, Luca Fiorito, Camila Braga Vieira, Simone Heusdains, Arien Baudoin, Carine Van Reybroeck, Margot Degrève, Antonín Krása, and Anatoly Kochetkov for making a nice and enjoyable atmosphere in the BR1 building.

Working at SCK•CEN as a PhD researcher, I was given the opportunity to supervise projects of several nuclear engineering students. The work of Antonios

Mylonakis, Iwein De Baetselier, and Clémence Bergeret and the help of Timothée Kooyman have been greatly beneficial for me and I would like to thank them for that.

I would also like to express my utmost appreciation to the SCK•CEN Scientific Council, which selected me as a PhD candidate in August 2009 and, by doing so, offered me the chance to enjoy the Flemish generosity and hospitality. My stay at SCK•CEN and KU Leuven was a very enriching experience. I value highly especially the extraordinary level of professionalism, research excellence, organisation, and overall culture at both world-class institutions.

A big thank you to Michèle Coeck, the head of the SCK•CEN Academy for Nuclear Science and Technology, for her kind patience.

Finally, I thank my parents Josef and Anna Jaluvka for providing me the best conditions for finalizing this dissertation and for their unconditional support. A substantial part of this text was written in Michigan in the United States. This was possible only thanks to the help and hospitality of David and Mary Baumgarten. Thank you Mr. and Mrs. Baumgarten. Lastly, I thank my fiancée Lindsey Rose Baumgarten for believing in me and for her love.

This research was performed in the framework of the SCK•CEN Academy PhD programme.

David Jaluvka
Mol, January 2015

Abstract

This dissertation develops a core management tool called RELOAD-M capable of optimizing reactor-core fuel loadings for MYRRHA, the future fast-spectrum research facility currently under development at SCK•CEN, Belgium. Such a tool is needed for designing highly efficient loading patterns that reflect various performance objectives of the multipurpose machine. RELOAD-M can solve the single-cycle loading pattern optimization problem, using different metaheuristic optimization methods and reactor analysis codes.

Two iterative population-based metaheuristics are implemented to solve the loading pattern optimization problem: Genetic Algorithm (GA) (with or without elitism) and Ant Colony Optimization (ACO). Both methods are applied to a simple core-reload problem with a known global optimum and the optimization results are compared. It is found that the elitist GA gives the most consistent results and performs best.

MYRRHA reactor-core models are described and used for the neutronics evaluation of different loading patterns by reactor analysis codes tailored to fast-spectrum systems. A simple thermal-hydraulics module is implemented for the calculation of the maximum fuel-cladding temperature. All employed models give results that are sufficiently accurate and fast enough for optimization purposes.

A MYRRHA loading pattern optimization problem is solved that aims at maximizing the facility's irradiation performance expressed in terms of the fast-neutron fluence achieved in reactor experimental channels. Three types of constraints are included in the problem: limited number of available fuel assemblies, maximum allowed fuel-cladding temperature, and end-of-cycle criticality condition. It is concluded that both the GA and ACO algorithms provide feasible solutions that outperform intuitively designed loading patterns. However, the resulting improvement is only marginal.

Samenvatting

Dit proefschrift beschrijft de onwikkeling van een rekencode, genaamd RELOAD-M, dat in staat is om de kernladingen voor MYRRHA the optimaliseren. MYRRHA is de toekomstige snel-spectrum onderzoeksinstallatie die momenteel ontwikkeld wordt door het SCK•CEN in België. RELOAD-M is nodig om efficiënte ladingsschema's te ontwerpen die voldoen aan de uiteenlopende doelstellingen van deze multifunctionele installatie. RELOAD-M kan de één-cyclus optimalisatieproblemen voor ladingsschema's oplossen door metaheuristische methodes en reactorcodes te gebruiken.

Twee iteratieve metaheuristische methodes, die gebaseerd zijn op populatie-technieken, zijn geïmplementeerd om het optimalisatievraagstuk voor de kernlading op te lossen: Genetic Algorithm (GA) (met of zonder elitarisme) en Ant Colony Optimization (ACO). Beide methodes zijn toegepast op een eenvoudig kernherladingsprobleem met een gekend globaal optimum en de resultaten zijn vergeleken. De GA methode met elitarisme levert de meest consistente resultaten en presteert het beste.

Verschillende modellen voor de MYRRHA reactorkernen zijn beschreven in dit werk. Deze modellen worden gebruikt in de reactoranalyse codes die gespecialiseerd zijn in het rekenen aan snel-spectrum systemen. Een eenvoudige thermohydraulische module is geïmplementeerd voor het berekenen van de maximale temperatuur van het omhulsel van de brandstofstaven. Alle gebruikte modellen leveren resultaten op die voldoende nauwkeurig en snel zijn voor de beoogde optimalisatiedoelstellingen.

Een specifiek MYRRHA optimalisatievraagstuk voor ladingsschema's is geanalyseerd, dat de bestralingsperformantie in de installatie moet maximaliseren. Deze bestralingsperformantie is gemeten in termen van de flux van snelle neutronen in de experimentele kanalen van de reactor. Er zijn drie restricties opgenomen in het vraagstuk: een gelimiteerd aantal beschikbare brandstofassemblages, een maximaal toegelaten temperatuur van het omhulsel van de brandstofstaaf

en de kritikaliteitsvoorwaarde aan het einde van de brandstofcyclus. Voor dit probleem leveren de GA en ACO algoritmen beide aanvaardbare oplossingen, die beter presteren dan intuïtief bepaalde ladingsschema's. De behaalde verbetering is echter beperkt.

Contents

List of Abbreviations

ABC	Artificial Bee Colony
ABCRK	Artificial Bee Colony with Random Keys
ABWR	advanced boiling water reactor
ACO	Ant Colony Optimization
ACS	Ant Colony System
ADONIS	Accelerator Driven Optimized Nuclear Irradiation System
ADS	accelerator driven system
ADTT	accelerator driven transmutation technology
AES	atomic energy station
AGM	alternative gradient method
AGR	advanced gas-cooled reactor
AI	artificial intelligence
ANL	Argonne National Laboratory
ANN	artificial neural network
AP	advanced passive (PWR)
AS	Ant System algorithm
ASCII	American Standard Code for Information Interchange
ATW	accelerator transmutation of waste
B3PEC	Branching and Bounding Batch Patterns Enumerated und Constraints
B&B	branch and bound
BA	Bat Algorithm
BE	Binary Exchange
BIMCMC	Bayesian interface along Markov Chain Monte Carlo
BOC	beginning of cycle
BOEC	beginning of equilibrium cycle
BOL	beginning of life

BP	burnable poison
BR1	Belgian Reactor 1
BR2	Belgian Reactor 2
BWAS	Best-Worst Ant System
BWR	boiling water-cooled and moderated reactor
CA	Cellular Automata
CBS	Class-Based Search
CBW	control bank worth
CCCC	Committee on Computer Code Coordination
CFA	Continuous Firefly Algorithm
CIA	cyclic interchange algorithm
CPO	collision probability method
CPR	critical power ratio
CPU	central processing unit
CR	control rod
CRP	Control Rod Programming
CSAS	Criticality Safety Analysis Sequence
CX	Cycle Crossover
DBA	design basis accident
DE	Differential Evolution
DERL	Differential Evolution with Random Localization
DFA	Discrete Firefly Algorithm
DGA	Distributed Genetic Algorithm
DHS	Differential Harmony Search
DLP	Dual Linear Programming
DP	Dynamic Programming
DPA	displacement per atom
DS	Direct Search
EA	Evolutionary Algorithm
ECCO	European Cell COde
EDA	Estimation of Distribution Algorithm
EFIT	European Facility for Industrial Transmutation
EFPD	effective full-power day
EHES	Elimination of Hopeless End States
ENDF	evaluated nuclear data file
ENEA	Italian National Agency for New Technologies, Energy and Sustainable Economic Development
EOC	end of cycle
EOEC	end of equilibrium cycle

EOL	end of life
EP	Evolutionary Programming
ERANOS	European Reactor ANalysis Optimized calculation System
ES	Evolution Strategy
ESES	Elimination of Similar End States
FA	Firefly Algorithm
FA	fuel assembly
FBR	fast breeder reactor
FMS	ferritic-martensitic stainless steel
FP	fission product
FPBIL	parameter free Population-Based Incremental Learning
FPF	flux-peaking factor
FR	fast reactor
GA	Genetic Algorithm
GCR	gas-cooled graphite-moderated reactor
GDA	Great Deluge Algorithm
GFR	gas-cooled fast reactor
GHS	Global-best Harmony Search
GIA	Gradient Information Algorithm
GIF	Generation IV International Forum
GP	Genetic Programming
GPM	gradient projection method
GUI	graphical user interface
HC	hill-climbing algorithm
HCMX	Heuristic Copy & Match Crossover
HER	hot excess reactivity
HLW	high-level waste
HM	heavy metal
HP	Haling principle
HS	Harmony Search
HTBX	Heuristic Tie-Breaking Crossover
IBA	Ion Beam Applications
ICFMOP	in-core fuel management optimization problem
IHS	Improved Harmony Search
ILP	Integer Linear Programming
IPPSO	Improved Pivot Particle Swarm Optimization
IPS	in-pile section

IX	Intersection Crossover
JEFF	Joint Evaluated Fission and Fusion File
KU Leuven	University of Leuven
LBE	lead-bismuth eutectic
LFCC	levelized fuel cycle cost
LFP	lumped fission product
LFR	lead-cooled fast reactor
linac	linear accelerator
LM	List Model
LMFBR	liquid-metal-cooled fast breeder reactor
LP	Linear Programming
LP	loading pattern
LPM	linear perturbation method
LPOP	loading pattern optimization problem
LWR	light-water-cooled and moderated reactor
MA	minor actinide
MAPRAT	maximum ratio of average planar linear heat generation rate
MARP	maximum average relative power
MCNP	General Monte Carlo N-Particle Transport Code
MCNPX	Monte Carlo N-Particle eXtended
MCPR	minimal critical power ratio
MFAB	maximum fuel assembly design
MFC	multifunctional channel
MFLCPR	maximum fraction of limiting critical power ratio
MFLPD	maximum fraction of linear power density
MILP	Mixed Integer Linear Programming
MINLP	Mixed Integer Nonlinear Programming
MINLP	mixed integer nonlinear program
MIP	Mixed Integer Programming
MLHGR	maximum linear heat generation rate
MMAS	Max-Min Ant System
MOC	method of characteristics
MOGA	multiobjective Genetic Algorithm
MOSA	multiobjective Simulated Annealing
MOX	mixed oxide
MpA	multi-point approximation
MPGR	maximum power generation ratio

MRNP	maximum relative nodal power
MSR	molten salt reactor
MTBF	mean time between failures
MTC	moderator temperature coefficient
MTR	material testing reactor
MYRRHA	Multi-purpose hYbrid Research Reactor for High-tech Applications
MYRRHA-FASTEF	MYRRHA FAst Spectrum Transmutation Experimental Facility
NEA	Nuclear Energy Agency
NLP	Nonlinear Programming
NP	non-deterministic polynomial-time
NRRCDP	nuclear reactor reload core design problem
NRRP	nuclear reactor reload problem
NSGA-II	Non-dominated Sorting Genetic Algorithm II
NSP	Nuclear Systems Physics
OECD	Organization for Economic Co-operation and Development
OX	Order Crossover
OX2	Order Crossover #2
PBIL	Population-Based Incremental Learning
PBX	Position Based Crossover
PDS-XADS	Preliminary Design Study of an eXperimental ADS
PhD	Doctor of Philosophy
PHWR	pressurized heavy-water-cooled and moderated reactor
PI	Pairwise Interchange
PICGA	Parallel Integer Coded Genetic Algorithm
PMX	Partially Mapped Crossover
P_N	method of spherical harmonics
PPF	power-peaking factor
PPP	pin peak power
PPSO	Parallel Particle Swarm Optimization
PSO	Particle Swarm Optimization
PSORK	Particle Swarm Optimization with Random Keys
PT	Perturbation Theory
PVAM	purposeful variant analysis method
PWR	pressurized water-cooled and moderated reactor

QACO	Quantum Ant Colony Optimization
QAE	Quantum Evolutionary Algorithm
QOI	quantity of interest
QPBIL	Quantum Population-Based Incremental Learning
QPSO	Quantum Particle Swarm Optimization
QPSO-DM	Quantum Particle Swarm Optimization—Differential Mutation
R&D	research and development
RAM	random-access memory
RAS	Rank-based Ant System
RELOAD-M	REactor Loading Optimization And Design for MYRRHA
RepU	reprocessed uranium
RNAH	Reactive Neighborhood Acceptance Heuristic
RNN	Recurrent Neural Network
RPPF	radial power-peaking factor
RR	research reactor
RRT	Record-to-Record Travel
RS	Random Search
RSICC	Radiation Safety Information Computational Center
SA	Simulated Annealing
SA	subassembly
SCALE	Standardized Computer Analyses for Licensing Evaluation
SCK•CEN	Belgian Nuclear Research Centre
SCWR	supercritical-water-cooled reactor
SDM	shutdown margin
SFR	sodium-cooled fast reactor
SGHS	Self-adaptive Harmony Search
SI	Swarm Intelligence
S_N	discrete ordinates method
SQPBIL	Self-adaptive Population-Based Incremental Learning
SR	safety rod
SS	stainless steel
STR	state transition rule
TBX	Tie-Breaking Crossover
TBX2	Tie-Breaking Crossover #2
T-H	thermal-hydraulics
TPSA	Temperature Parallel Simulated Annealing

TRU	transuranium element
TS	Tabu Search
TSP	Traveling Salesman Problem
UML	Unified Modeling Language
UX	Union Crossover
UX2	Union Crossover #2
VARIANT	VARIational Anisotropic Nodal Transport for Multidimensional Cartesian and Hexagonal Geometry Calculation
VEPSO	Vector Evaluated Particle Swarm Optimization
VHTR	very-high-temperature reactor
VOCT	Variational Optimal Control Theory
WFM	weighting factor method
XLHGR	linear heat generation rate
XMCPR	minimal critical power ratio
XML	Extensible Markup Language
XMPGR	fraction of the limiting average linear generation rate
XT-ADS	eXperimental facility demonstrating the technical feasibility of Transmutation in an Accelerator Driven System
ZCU	zone controller unit

List of Symbols

A	transport (destruction) operator from the Boltzmann equation (2.20) on p. 35
A_{cor}	corner subchannel area (m^2)
A_{edg}	edge subchannel area (m^2)
A_{inn}	inner subchannel area (m^2)
A_{tot}	total subchannel area (m^2)
B	fuel burnup ($\mathrm{MW\,d\,t_{HM}^{-1}}$)
B_{c}	fuel cycle burnup ($\mathrm{MW\,d\,t_{HM}^{-1}}$)
B_{d}	fuel batch discharge burnup ($\mathrm{MW\,d\,t_{HM}^{-1}}$)
C	conversion ratio
$C(k,n)$	number of k-combinations from a given set of n elements
c_{B}	soluble boron concentration (ppm)
$c_{\mathrm{p\,LBE}}$	LBE heat capacity ($\mathrm{J\,kg^{-1}\,{}^{\circ}C^{-1}}$)
D	fuel pin outer diameter (m)
$D(\boldsymbol{r}, E)$	neutron diffusion coefficient (cm)
$D_g(\boldsymbol{r})$	neutron diffusion coefficient in energy group g (cm)
$\mathbb{D}_g(\boldsymbol{r})$	directional neutron diffusion coefficients in energy group g (cm), $\mathbb{D}_g(\boldsymbol{r}) \in \mathbf{R}^3 \times \mathbf{R}^3$
$d_{\mathrm{h\,edg}}$	edge subchannel hydraulic diameter (m)
$d_{\mathrm{h\,inn}}$	inner subchannel hydraulic diameter (m)
$d_{\mathrm{h\,cor}}$	corner subchannel hydraulic diameter (m)
E	neutron energy (eV)
F	fission (production) operator from the Boltzmann equation, see Eq. (2.20) on p. 35

$\mathcal{F}$	set of feasible solutions
F_b	fraction of the fuel atoms that undergo fission in an FBR, Eq. (1.3) on p. 4
F_Q	nuclear heat flux hot channel factor
$F_{\Delta H}$	enthalpy hot channel factor
$f(\boldsymbol{x})$	objective function
$f_\mathrm{a}(\boldsymbol{x})$	augmented objective function
f_SCH	subchannel mass-flow rate peaking factor
G	maximum number of generations
g	generation
$g(\boldsymbol{x})$	constraint function
$\mathcal{H}$	heuristic information used in ACO
h	heat transfer coefficient ($\mathrm{W\,m^{-1}\,{}^\circ C^{-2}}$)
$h(\boldsymbol{r}, E)$	power factor from Eq. (2.22) on p. 36 ($\mathrm{J\,n^{-1}}$)
$h_g(\boldsymbol{r})$	power factor for the energy group g from Eq. (2.33) on p. 39 ($\mathrm{J\,n^{-1}}$)
I	number of depletion steps
I	maximum number of iterations
$\mathcal{I}$	interface between two adjacent regions
i	iteration
$\boldsymbol{J}(\boldsymbol{r}, E, t)$	(angle-)integrated neutron current ($\mathrm{n\,cm^{-2}\,eV^{-1}\,s^{-1}}$)
$\boldsymbol{J}(\boldsymbol{r}, E, \boldsymbol{\Omega}, t)$	angular neutron current ($\mathrm{n\,cm^{-2}\,eV^{-1}\,s^{-1}}$)
$J_g^-(\boldsymbol{r})$	incoming net neutron current for an energy group g ($\mathrm{n\,cm^{-2}\,s^{-1}}$)
$J_g^+(\boldsymbol{r})$	outgoing net neutron current for an energy group g ($\mathrm{n\,cm^{-2}\,s^{-1}}$)
k	parameter in the adaptive penalty factor from Eq. (4.26), p. 154
k	ant population size
k_eff	effective multiplication factor
$k_\mathrm{eff}^\mathrm{EOC}$	EOC effective multiplication factor
k_eff^*	effective multiplication factor obtained as a result of optimization
k_LBE	LBE thermal conductivity ($\mathrm{W\,m^{-1}\,{}^\circ C^{-1}}$)
k_ox	oxide layer thermal conductivity ($\mathrm{W\,m^{-1}\,{}^\circ C^{-1}}$)
k_src	subcritical source-multiplication factor
k_∞	infinite multiplication factor

L	anisotropic scattering order
L	losses during reprocessing and fuel fabrication, Eq. (1.3) on p. 4
L_{act}	active length (m)
L_{ext}	extrapolated length (m)
LMN	DIF3D-VARIANT nodal spatial approximation orders (L—source, M—flux, N—leakage)
$\boldsymbol{l}$	vector representation of a refueling scheme consisting of n batches of given sizes, $\boldsymbol{l} = (l_1, l_2, \ldots, l_n)$
M	number of available fuel assemblies
MN	DIF3D-VARIANT nodal angular approximation orders (M—flux, N—leakage)
m	number of constraints
$\boldsymbol{m}$	vector representation of an available fuel-assembly inventory consisting of N fuel types, $\boldsymbol{m} = (m_1, m_2, \ldots, m_N)$
m_{core}	core mass flow rate ($\text{kg}\,\text{s}^{-1}$)
m_{FA}	average fuel assembly mass flow rate ($\text{kg}\,\text{s}^{-1}$)
m_{inn}	inner subchannel mass flow rate ($\text{kg}\,\text{s}^{-1}$)
m_{pin}	average pin mass flow rate ($\text{kg}\,\text{s}^{-1}$)
N	number of fuel-assembly types
Nu	Nusselt number
$\boldsymbol{N}(\boldsymbol{r})$	unit vector normal to a given surface at point $\boldsymbol{r}$
$N(t)$	nuclide concentration (cm^{-3})
N_{cor}	number of corner subchannels in a fuel assembly
N_{edg}	number of edge subchannels in a fuel assembly
N_{FA}	number of fuel assemblies in a reactor core
N_{fiss}	number of fissonable nuclides
N_{g}	number of energy groups
N_{inn}	number of inner subchannels in a fuel assembly
N_{nucl}	number of considered nuclides
N_{pin}	number of pins in a fuel assembly
N_{SCH}	number of subchannels in a fuel assembly
$\mathbf{N}_0$	set of whole (non-negative natural) numbers
n	number of fuel batches in an equilibrium core
n	number of variables
$\mathcal{O}$	set of children
o	number of children produced per one parents set

P	fuel pin pitch (m)
P	core power (MW); thermal (MW_{th}) or electric (MW_e)
$\mathcal{P}$	set of loading patterns (population)
Pe	Péclet number
Pr	Prandtl number
$P(k,n)$	number of k-permutations from a given set of n elements
$P(r,s,p)$	STR probability-distribution function from Eq. (4.5) on p. 124
$P_l(\mu)$	l-order Legendre polynomial
P_{lim}	core power limit (MW)
P_{q}	(total) power-peaking factor
P_{r}	radial power-peaking factor
$P_{\text{r core}}$	whole-core per-assembly component of the radial power-peaking factor
$P_{\text{r FA}}$	inner-assembly component of the radial power-peaking factor
P_{t}	modified pitch for spacing wire pressure drop correlation (m)
P_{z}	axial power-peaking factor
p	population size (individuals)
p	position variable in the ACO algorithm
p_{c}	crossover rate (probability)
p_{f}	power fraction
p_{m}	mutation rate (probability)
$Q(\boldsymbol{r},E,\boldsymbol{\Omega},t)$	neutron source term ($\text{n}\,\text{cm}^{-3}$)
$Q_g(\boldsymbol{r},\boldsymbol{\Omega})$	neutron source term for energy group g ($\text{n}\,\text{cm}^{-3}$)
$\bar{q}$	core average linear power ($\text{W}\,\text{cm}^{-1}$)
$q(\boldsymbol{r},E,t)$	external neutron source term ($\text{n}\,\text{cm}^{-3}$)
$\bar{q}_g^k$	node-averaged power density at node k corresponding to the neutron flux of energy group g ($\text{W}\,\text{cm}^{-3}$)
$\bar{q}_{\text{hotFA}}$	average linear power in a hot fuel assembly ($\text{W}\,\text{cm}^{-1}$)
q_{max}	maximum linear power ($\text{W}\,\text{cm}^{-1}$)
q_0	parameter from the ACO algorithm
R	number of rings of LBE haxagons outside of the core barrel
$\mathbf{R}$	set of real numbers
$\mathbf{R}^-$	set of negative real numbers

$\mathbf{R}^+$	set of positive real numbers
Re	Reynolds number
$R_l^m(\boldsymbol{\Omega})$	l-order real spherical harmonics component
r	number of integer variables in a MINLP (2.8)–(2.11), p. 24
$\boldsymbol{r}$	spatial vector, $\boldsymbol{r} \in \mathbf{R}^3$
$\mathcal{S}$	set of parents
s	number of parents sets
$\boldsymbol{s}$	vector of fuel-cell frequencies, $\boldsymbol{s} = (s_1, s_2, \ldots, s_{N_{\mathrm{FA}}})$
$T_{1/2}$	half-life (s)
T	temperature (K)
$\mathcal{T}$	pheromone model
T_{clad}	maximum outer fuel-cladding temperature (°C)
$T_{\mathrm{clad}}^{\mathrm{lim}}$	maximum outer fuel-cladding temperature limit (°C)
$T_{\mathrm{hotSCH}}(z)$	maximum hot subchannel fuel-cladding temperature (°C)
T_{in}	average coolant inlet temperature (°C)
t	time (s)
$\boldsymbol{t}$	vector representation of N fuel-assembly types ordered in an ascending order, $\boldsymbol{t} = (t_1, t_2, \ldots, t_N)$
t_{c}	fuel cycle length (d)
U_{e}	uranium utilization efficiency
u	lethargy; $u = \ln \frac{E_0}{E}$, where E_0 is a constant
V	reactor volume (m^3)
V_{IPS}	volume of the IPS region in the core fissile zone (m^3)
$v(E)$	neutron speed ($\mathrm{m\,s}^{-1}$)
v_{cor}	corner subchannel coolant speed ($\mathrm{m\,s}^{-1}$)
v_{edg}	edge subchannel coolant speed ($\mathrm{m\,s}^{-1}$)
v_{inn}	inner subchannel coolant speed ($\mathrm{m\,s}^{-1}$)
x	fuel enrichment (wt% of HM)
x	objective function value (fitness)
$\boldsymbol{x}$	vector of n decision variables $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T$
$\boldsymbol{x}$	vector representation of a loading pattern consisting of N_{FA} fuel assemblies $\boldsymbol{x} = (x_1, x_2, \ldots, x_{N_{\mathrm{FA}}})^T$
x_{f}	feed enrichment (wt% of HM)

$\boldsymbol{x}_{\text{iter}}$	iteration-best solution
x_{opt}	best objective-function value
x_{p}	product enrichment ($x_{\text{p}} \equiv x$) (wt% of HM)
x_{t}	tails enrichment (wt% of HM)
$\boldsymbol{x}_{\text{opt}}$	vector of optimum decision variables or the best loading-pattern vector
$\boldsymbol{y}$	vector of loading pattern characteristics
$\mathbf{Z}$	set of integer numbers
Z_{hotSCH}	axial position of the maximum hot subchannel fuel cladding temperature (m)
z	axial position/direction (m)

Greek Symbols

α	fission to capture microscopic cross section ratio
α	scaling penalty factor
β	albedo
$\beta(t)$	total delayed neutron fraction
$\beta_g(\boldsymbol{r})$	albedo in energy group g
$\chi(E)$	neutron fission spectrum
$\chi_{i\,g}(\boldsymbol{r})$	neutron fission spectrum in energy group g for fissile nuclide i
$\delta_{i\,j}$	Kronecker delta
δ_{ox}	oxide-layer thickness (m)
ϵ	infinitesimally small positive real number, $\epsilon \in \mathbf{R}$
η	heuristic information value, $\eta \in \mathcal{H}$
γ	parameter from the ACO global updating rule (4.7)
κ	elitist group size
λ	decay constant (s^{-1})
$\lambda_{i,j}$	decay constant for the decay from nuclide j to nuclide i (s^{-1})

μ	particle direction cosine
μ	mean value
μ	tournament selection size
$\bar{\mu}(\boldsymbol{r})$	average cosine of the neutron deviation angle in a collision
ν	size of a parents set $\mathcal{S}$, $\nu = \lvert\mathcal{S}\rvert$
ν_i	number of neutrons produced per fission of the i-th fissionable nuclide
$\nu_{i\,g}(\boldsymbol{r})$	average number of neutrons produced per fission of fissile nuclide i in energy group g
$\boldsymbol{\Omega}$	angular vector, $\boldsymbol{\Omega} \in \mathbf{R}^2$
ω	weighting penalty factor
$\phi(\boldsymbol{r}, E, t)$	(angle-integrated) neutron flux ($\mathrm{n\,cm^{-2}\,eV^{-1}\,s^{-1}}$)
$\phi_g(\boldsymbol{r})$	neutron flux in energy group g ($\mathrm{n\,cm^{-2}\,s^{-1}}$)
$\bar{\phi}_g^k$	node-averaged value of the neutron flux for energy group g at node k ($\mathrm{n\,cm^{-2}\,s^{-1}}$)
$\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t)$	angular neutron flux ($\mathrm{n\,cm^{-2}\,sr^{-1}\,eV^{-1}\,s^{-1}}$)
$\psi^-_{\partial V}$	incoming angular neutron flux on outer boundary ∂V ($\mathrm{n\,cm^{-2}\,sr^{-1}\,s^{-1}}$)
ρ	pheromone evaporation rate
$\rho(t)$	reactivity
ρ_{LBE}	LBE density ($\mathrm{kg\,m^{-3}}$)
$\Sigma_{\mathrm{a}}(\boldsymbol{r}, E)$	macroscopic absorption cross section ($\mathrm{cm^{-1}}$)
$\Sigma_{\mathrm{f}}(\boldsymbol{r}, E)$	macroscopic fission cross section ($\mathrm{cm^{-1}}$)
$\Sigma_{g \leftarrow h}(\boldsymbol{r})$	macroscopic scattering cross section from group h to group g ($\mathrm{cm^{-1}}$)
$\Sigma_{\mathrm{f}\,i\,g}(\boldsymbol{r})$	macroscopic fission cross section for fissile nuclide i in energy group g ($\mathrm{cm^{-1}}$)
$\Sigma_{\mathrm{s}}(\boldsymbol{r}, E \leftarrow E', \boldsymbol{\Omega} \leftarrow \boldsymbol{\Omega}')$	macroscopic double differential scattering cross section ($\mathrm{cm^{-1}}$)
$\Sigma_{\mathrm{s}\,g}(\boldsymbol{r})$	macroscopic scattering cross section in energy group g ($\mathrm{cm^{-1}}$)
$\Sigma_{\mathrm{t}}(\boldsymbol{r}, E, \boldsymbol{\Omega}, t)$	macroscopic total cross section ($\mathrm{cm^{-1}}$)
$\Sigma_{\mathrm{t}\,g}(\boldsymbol{r})$	macroscopic total cross section in energy group g ($\mathrm{cm^{-1}}$)
$\Sigma_{\mathrm{tr}\,g}(\boldsymbol{r})$	macroscopic transport cross section in energy group g ($\mathrm{cm^{-1}}$)

Σ_{x}	macroscopic cross section for reaction x (cm^{-1})
σ	standard deviation
σ_{a}	microscopic absorption cross section (b)
$\sigma_{\mathrm{c}}(E)$	microscopic capture cross section (b)
$\sigma_{\mathrm{f}}(E)$	microscopic fission cross section (b)
$\sigma_{i,j}(\boldsymbol{r}, E)$	particle-induced microscopic cross section for transmutation from nuclide j to nuclide i (b)
σ_x	microscopic cross section for reaction x (b)
τ	pheromone value, $\tau \in \mathcal{T}$
τ_0	initial pheromone value
$\Theta(\boldsymbol{r}, E)$	neutron fluence ($\mathrm{n\,cm^{-2}\,eV^{-1}}$)
$\bar{\Theta}_{\mathrm{IPS}}^{\mathrm{fast}}$	average (energy-integrated) fast-neutron fluence in IPSs channels ($\mathrm{n\,cm^{-2}}$)
ζ	gene exchange probability in the uniform crossover operator

Other Symbols

$\uparrow$	maximum value of
$\nearrow$	maximize (e.g., function)
$\downarrow$	minimum value of
$\searrow$	minimize (e.g., function)
∂V	spatial boundary (surface) of the volume V

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

The first chapter of the dissertation explains the motivation for the presented work and summarizes its main objectives. The organization of the text is briefly described at the end of the chapter.

1.1 Motivation

This section tries to answer the basic question "Why is this work needed?" or "What is it useful for?" The answer is presented in a way understandable for all readers by using the general-to-specific approach, that is, beginning with some general aspects of nuclear power generation, continuing with the emerging need for advanced nuclear reactor types and fuel cycles, the MYRRHA project, and ending with the actual problem of the MYRRHA reactor core management and required analytical capabilities.

1.1.1 Nuclear Power

Nuclear power was responsible for 11 % of all electricity generated in the world at the end of 2012. The total number of operating power reactors was 437. Out of them, 357 were light-water-cooled and moderated reactors (LWRs), that is pressurized water-cooled and moderated reactors (PWRs) and boiling water-cooled and moderated reactors (BWRs), representing 89 % of the total net electrical power. Only two operating units were reactors with fast neutron spectrum, whereas the rest of the units were thermal spectrum reactors. 67

new units were under construction in 2012 out of which 58 were LWRs (IAEA, 2013a). All currently operating reactors utilize enriched or natural UO_2 fuel. Only a fraction of the spent fuel is reprocessed and reused again in the form of mixed oxide (MOX) or reprocessed uranium (RepU) fuel. The use of MOX and RepU fuel accounted for 1126 respectively 899 tonnes of equivalent natural uranium in 2010, which represented tiny amounts comparing to the 63 875 t_U of annual natural uranium requirements claimed in the same year (OECD-NEA, 2012).

Although nuclear power has proven itself as a reliable energy source with more than 15 000 years of operating experience (IAEA, 2013b), its public perception has been severely damaged in the past as a consequence of three major nuclear accidents—the Three Mile Island accident (Pennsylvania, USA, 1979), the Chernobyl disaster (Ukraine, USSR, 1986), and the most recent Fukushima Daiichi nuclear disaster (Japan, 2011). Another important aspect, which makes nuclear power a less favorable energy solution, is the radioactive waste production. Despite its relatively small volume[1], the spent fuel[2] inventory remains highly radiotoxic for hundreds of thousands of years before it drops to the level of natural uranium, and thus, places a considerable burden upon future generations.

In order to make nuclear power more attractive again, the above mentioned weaknesses (besides others such as an economical competitiveness) need to be addressed while maintaining or increasing current safety levels and limiting proliferation risks. This would be possible only by revising the whole concept of the present nuclear energy production, which almost solely relies on thermal spectrum reactors and the "once-through" uranium fuel cycle[3].

1.1.2 Advanced Nuclear Systems

The need for a principal shift in the persisting nuclear energy paradigm was recognized by a group of nine leading nuclear countries which joined together in 2000 to form the Generation IV International Forum (GIF). The aim of the

[1] Approximately 180 000 t_{HM} of spent fuel accumulated worldwide was reported in May 2006 (IAEA, 2007).

[2] Spent nuclear fuel is nuclear fuel that has been irradiated in a nuclear reactor. It is composed of different constituent nuclides, among them fission products (FPs) and minor actinides (MAs) built up through fission respectively radiative capture reactions on the nuclides originally present in the fresh fuel (see Figure 3.5 in Chapter 3, for instance). FPs and MAs are responsible for the radiotoxicity of spent fuel and contribute to deterioration of reactivity, a measure of criticality defined as $\rho = (k_{\text{eff}} - 1) \times k_{\text{eff}}^{-1}$.

[3] An alternative solution that would relax the problem of the scarce uranium resources could be utilization of the thorium fuel cycle. This option, however, will not be discussed in this work as it is concerned only with the uranium cycle.

forum is to develop next-generation nuclear energy systems that would solve the issues associated with the existing nuclear park and enhance the future role of nuclear energy (GIF, 2002).

Generation IV. Nuclear systems can be categorized into several generations according to their technological maturity and the time of their deployment. The early prototype reactors deployed in the 1950s and 1960s are dubbed the first generation. The second generation began in the 1970s in the large commercial power plants that are still operating today. Generation III was developed more recently in the 1990s with a number of evolutionary designs that offer significant advances in safety and economics. The most recent Generation III+ includes state-of-the-art designs currently being deployed around the world. Examples of such designs are AP1000, AES-2006, or ABWR (GIF, 2002, 2014).

In 2002, GIF selected six revolutionary types of reactors with innovative fuel cycle technologies with the aim to respond to the following main sustainability criteria and future market conditions: to incorporate advanced nuclear safety; to be resistant to proliferation in addressing nuclear non-proliferation and physical protection against aggression; to be highly economic and competitive; and to minimize waste production and optimize natural resource utilization. The six most promising technologies collectively known as Generation IV nuclear energy systems include gas-cooled fast reactor system (GFR), lead-cooled fast reactor system (LFR), molten salt reactor system (MSR), sodium-cooled fast reactor system (SFR), supercritical-water-cooled reactor system (SCWR), and very-high-temperature reactor system (VHTR). Three of the proposed Generation IV designs, SFR, GFR, and LFR, are fast spectrum reactors with closed fuel cycles. The Generation IV systems may be available for commercial application around 2030–2040.

Fast Reactors

As the name suggests, fast reactors are characterized by neutrons that are moving substantially faster in a reactor core than neutrons in thermal reactor types. This gives fast reactors unique properties and makes them prominent designs from the viewpoint of the fuel utilization and waste reduction. The harder neutron spectrum in fast reactors is achieved by avoiding the use of coolants made of atoms with light nuclei like hydrogen, making the process of neutron slowing down (moderation) less efficient than water cooled (and moderated) reactors, for instance.

The primary interest in fast reactors has always been tied to their ability to greatly extend fuel supplies. In thermal reactors, neutrons fission primarily

^{235}U, which is present at $x_\mathrm{f} \approx 0.71\,\mathrm{wt\%}$ in feed natural uranium, and only a small part of the more abundant fertile isotope ^{238}U can be either fissioned directly or converted into higher fissile nuclides like ^{239}Pu:

$$^{238}\mathrm{U}(\mathrm{n},\,\gamma)^{239}\mathrm{U} \xrightarrow[23.5\,\mathrm{min}]{\beta^-} {}^{239}\mathrm{Np} \xrightarrow[2.36\,\mathrm{d}]{\beta^-} {}^{239}\mathrm{Pu}\ . \tag{1.1}$$

The ratio U_e of material utilized to produce energy to total fuel material mined can be approximated by

$$U_\mathrm{e} = x_\mathrm{p}\frac{x_\mathrm{f} - x_\mathrm{t}}{x_\mathrm{p} - x_\mathrm{t}}\frac{1}{1 - C}\ , \tag{1.2}$$

where x_p is the product enrichment, x_t is the tails enrichment, and C is the conversion ratio defined as the ratio between fissile atoms produced by conversion and fissile atoms consumed. For the values representative for LWR, $x_\mathrm{p} = 4.50\,\mathrm{wt\%}$, $x_\mathrm{t} = 0.25\,\mathrm{wt\%}$, and $C = 0.6$, Eq. (1.3) yields a fuel utilization factor of only 1.2 %.

Substantially larger values of U_e can be obtained in case of fast reactors. For specific designs, a value of C can exceed unity, meaning that more usable material is produced during the operation than consumed. Fast reactors with $C > 1$ are called fast breeder reactors (FBRs). If one defines F_b as the fraction of the fuel atoms (fissile or fertile) fed to the reactor that undergoes fission during the time the fuel resides in the reactor, and L as the losses during reprocessing and fuel fabrication, then the fuel utilization for a FBR may be expressed as follows:

$$U_\mathrm{e} = \frac{F_\mathrm{b}}{F_\mathrm{b} + L}\ . \tag{1.3}$$

Hence, an FBR of $F_\mathrm{b} = 0.03$ and $L = 0.01$ has a potential utilization in the vicinity of 75 %, a significant improvement over the value estimated for a light water reactor (Graves, 1979).

Other advantages attributed to a fast neutron spectrum are the favorable fission-to-capture cross section ratios α for minor actinides, substances that are together with Pu responsible for the bulk of the long-term radiotoxicity and heat production of used nuclear fuel. The the probability that a neutron absorbed in a minor actinide nucleus will cause its fission is much larger for fast reactors than for LWRs, making fast reactors a suitable technology for nuclear high-level waste (HLW) incineration.

Indeed, the inherent capability of fast reactors to substantially alleviate issues associated with the nuclear waste has been deeply investigated and confirmed in many studies (e.g., IAEA, 2009; OECD-NEA, 2002, 2006). Two fundamental solutions to the HLW problem have been identified that are based on the fast

reactor technology[4]. One of them assumes an extensive deployment of critical reactors as the content of a transuranium element (TRU) with a high minor actinide share in the fuel must stay limited. The reason for that is that larger concentrations of minor actinides in a critical fast reactor core would lead to a deterioration of some important safety and operability parameters (OECD-NEA, 2002).

Therefore, an alternative fuel cycle strategy has been proposed that would allow for a greater flexibility with respect to the fuel composition and, at the same time, enhance safety. In this approach, known as the "double strata" approach, reactors are sorted into two categories or strata: the first stratum consists of reactors (thermal or fast) that are devoted to power generation using U- or Pu-based fuel, whereas the second stratum consists of dedicated fast spectrum transmuters loaded with a high minor actinide content. The most often mentioned candidate transmutation technology for the second stratum are subcritical reactors better known as accelerator driven systems.

Accelerator Driven Systems

Accelerator driven systems (ADSs)[5] are composed of three essential parts coupled together: a subcritical core, a spallation target, and a particle accelerator.

Supposing that there is a certain number of neutron induced fissions in a reactor at a certain time. Unlike in critical cores, where this number stays constant in time during operation, in subcritical cores without an external neutron source this number would be progressively decreasing (dying out) until there are no more neutrons causing fissions. The criticality of a reactor is usually expressed in terms of the effective multiplication factor k_{eff}, which can be explained as

$$k_{\text{eff}} = \frac{\text{number of fissions in generation } i}{\text{number of fissions in generation } i-1} \ . \tag{1.4}$$

For critical reactors $k_{\text{eff}} = 1$ and for subcritical reactors $k_{\text{eff}} < 1$. Reactors with $k_{\text{eff}} > 1$ are called supercritical.

In order to sustain the chain reaction in a subcritical core, an external source of neutrons is needed that compensates for the insufficient neutron balance caused either by sterile absorptions of the neutrons or by their excessive leakage out of the system. In an ADS, the supplementary neutrons are produced by spallation reactions, during which high energy protons, coming from a particle accelerator, are hitting a spallation target made of heavy metal. In the course

[4]Another future solution would be so-called fission-fusion hybrid systems (IAEA, 2009).

[5]ADSs are also known as accelerator driven transmutation technologies (ADTTs), accelerator transmutation of wastes (ATWs) systems, or simply hybrid systems.

of the spallation process, the liquid metal nuclei emit a large number of high energy neutrons. For example, about 13 primary neutrons are released per one 600 MeV proton during a spallation reaction on a lead-bismuth target. The energy spectrum of the spallation neutrons (i.e., neutrons that are released in a spallation reaction) ranges in such a case from keV energies up to the beam energy[6].

The concept of subcriticality allows to design pure TRU burners (i.e., $C = 0$), leading to transmutation performances unreachable in critical fast reactors mainly due to potential safety related problems. The absence or a very low content of uranium combined with a high content of minor actinides produces, as a consequence, a very low Doppler reactivity coefficient[7] (in general, mostly attributed to neutron capture rates in ^{238}U), and a very low fraction of delayed fission neutrons (Tsvetkov et al., 2012).

It is the enhanced safety and operability characteristics that give a subcritical burner a better capability to accept U-free fuels and to reach higher transmutation performances than in a critical fast reactor (see, e.g., Romanello et al., 2011).

1.1.3 MYRRHA

True to its tradition of a leading nuclear research institute, SCK • CEN is working since several years at the pre- and conceptual design of a multi-purpose flexible irradiation facility that can replace the BR2 material testing reactor (MTR), the current flagship of the centre's infrastructure, and that is innovative to pursue future oriented research projects. Motivated by the research needs in support of the development of next generation nuclear systems, design choices were made that led to what is now known as the Multi-purpose hYbrid Research Reactor for High-tech Applications (MYRRHA) reactor (Aït Abderrahim, Al Mazouzi, et al., 2012; Aït Abderrahim, De Bruyn, Van den Eynde, et al., 2014; Baeten et al., 2011).

MYRRHA is conceived as a fast spectrum liquid-metal-cooled reactor, able to operate in both subcritical and critical mode. As such, MYRRHA will play a role of an ADS and LFR technology demonstrator. The flexible design

[6] More than 1.5 % of the spallation neutrons from this example have energies larger than 200 MeV, which is approximately equal to the fraction of spallation neutrons with energies lower than 200 keV. with the peak value between 2 and 3 MeV.

[7] The Doppler reactivity coefficient expresses the change in reactivity in the nuclear fuel per degree change in the fuel temperature, due to the Doppler effect caused by a temperature-dependent distribution of velocities of atoms or molecules relative to the neutrons. Its typical values are from $-2\,\mathrm{pcm\,K^{-1}}$ to $-3\,\mathrm{pcm\,K^{-1}}$ for thermal reactors and around $-0.2\,\mathrm{pcm\,K^{-1}}$ for MYRRHA-like fast reactors (Bubelis and Schikorr, 2012; Reuss, 2008).

and neutronics performance of the reactor core will allow many applications, including HLW transmutation research or fuel and material development for Generation IV systems and fusion reactors[8].

The MYRRHA project involves a complete set of ongoing research and engineering activities needed for its successful realization. Among them, reactor systems and accelerator design and development, material research, or safety analyses. It is often the case that new, unconventional approaches have to be adopted in the R&D process in order to reflect the special nature of a first-of-a-kind facility. This is also true in case of modeling tools and methodologies used for the MYRRHA reactor core design and fuel management.

1.1.4 MYRRHA Core Management Tool

The research work presented in this dissertation is a part of a broader group of undergoing R&D activities aimed at development of a set of dedicated modeling and analytical tools and methodologies in support of MYRRHA reactor core design, operation, and nuclear fuel management.

In the past, the Monte Carlo depletion code ALEPH (Haeck and Verboomen, 2007; Stankovskiy and Van den Eynde, 2011; Stankovskiy, Van den Eynde, and Vidmar, 2011) was developed in the NSP expert group at SCK•CEN, a coupling of the steady-state Monte Carlo transport code MCNP (X-5 Monte Carlo Team, 2003) or MCNPX (X-5 Monte Carlo Team, 2008), and a depletion solver RADAU, based on implicit Runge-Kutta methods (Hairer and Wanner, 1980).

ALEPH is used as a reference tool for MYRRHA reactor core design evaluations comprising neutronics calculations and fuel cycle analysis. As any Monte Carlo code, ALEPH suffers from two drawbacks, which make its use for routine fuel management calculations somewhat cumbersome. First, the probabilistic method imposes a significant computational burden and, second, obtained results are typically local results, that is, results describing a relatively small localized part of the modeled system. In contrary to this, as will be explained in Section 2.1, the process of selecting a loading pattern for each new fuel cycle requires evaluation of different global characteristics of plentiful reactor core configurations. Therefore, a need emerged for a new complementary reactor analysis framework that will be sufficiently fast, computationally inexpensive, and able to deliver required results.

[8]See Section 2.4 in Chapter 2 for more information regarding MYRRHA.

1.2 Research Goals

The goal of this work was to develop a modeling framework with an optimization capability that could be used for rapid evaluation of different core configurations and other scoping analyses to be performed within the MYRRHA nuclear fuel management.

Such a MYRRHA core management tool should be able to calculate in a reasonable time an optimal core given a stock pile of fresh and partially used fuel assemblies, while reflecting the special character of the facility. Therefore, it was necessary to investigate an optimality criterion that should be consistent with the specific purpose of the facility and to identify proper neutron transport methods to be used for the evaluation of the core characterized by many unique design features.

1.3 Dissertation Synopsis

This dissertation is focused on the use of proper nuclear fuel management techniques for MYRRHA core reload optimization and preliminary quantitative analysis. The following chapters of the dissertation cover in detail topics on analytical approaches used in nuclear fuel management, application of these approaches to specific MYRRHA fuel management problems, and their practical implementation.

Chapter 2 reviews the concept of nuclear fuel management to give a reader better understanding of the general context of the work. A special emphasis is placed on the problem of nuclear reactor loading pattern optimization and computational methods used for its solution. A separate section on MYRRHA is also included in the chapter with general information on the project and the reactor technology.

Chapter 3 details models developed for the purpose of MYRRHA core loading optimization. It contains a description of the geometry and material composition of the modeled system. The reactor analysis codes used to evaluate these models are also described in this chapter. The developed models are then tested on given core designs.

Chapter 4 describes various MYRRHA in-core fuel management problems that were solved using different optimization methods, while employing the models from the previous part. Chapter 4 also includes a detailed description of the applied optimization methods and a discussion of the obtained results.

Chapter 5 presents the RELOAD-M optimization framework used to solve the MYRRHA in-core fuel management optimization problems. The design and functionality of the core management tool are described and explained.

Chapter 6 summarizes the overall conclusions and recommendations of the dissertation. The results obtained are reviewed and the implications discussed. Several paths for future research are suggested as well.

Nuclear engineers and other readers mainly interested in the MYRRHA neutronics models developed during this PhD project are advised to skip to Chapter 3. The computer science audience and readers interested in the optimization part of the dissertation are recommended to skip to Chapter 4. Chapter 2 can be read by all readers as an introductory text to both research areas—neutroncis analysis and optimization.

1.4 Summary

Nuclear power has been successfully produced since 1950s and it has proven to be a reliable source of power. Current nuclear power stations are characterized by poor fuel utilization and production of an unnecessary large amount of highly radiotoxic waste. Therefore, a significant change is needed in the nuclear fuel cycle that would make nuclear power a more sustainable and versatile energy source.

Six Generation IV nuclear systems have been proposed that would alleviate the issues associated with the existing nuclear fleet. Three of them are critical fast spectrum reactors which can, in principle, utilize the whole potential of the fuel. However, the performance of critical fast reactors as MA burners is limited mainly due to problems related to reactor control. A higher transmutation performance can be achieved in subcritical nuclear systems.

Although the ADS concept has been known for decades, it has never been realized because of many technological challenges. It is only recently that the interest in ADS has been renewed. One of the ambitious projects that could demonstrate the viability of the concept and provide a platform for a research related to Generation IV systems is MYRRHA.

MYRRHA is currently under development at SCK•CEN in Mol, Belgium. It is conceived as a fast neutron irradiation facility that can be operated in a sub-critical or in a critical mode. The development of the facility requires many original engineering solutions. Considerable efforts have been made in the development of proper neutronics modeling tools that resulted in the Monte

Carlo burnup code ALEPH. The next step is the development of a framework for MYRRHA core management optimization—the objective of this work.

Chapter 2

Background

The purpose of this chapter is to put the presented work into a general context and to provide a supporting rationale for the problem solving approaches adopted in the later chapters. It begins with a section on nuclear fuel management followed by a separate part on the loading pattern optimization problem, the main subject of the dissertation. A special section is dedicated to practices used within nuclear fuel management for nuclear reactor core analysis. The chapter is then closed with some information on the MYRRHA project and the summary. An emphasis is placed on fast reactors throughout the whole chapter.

2.1 Nuclear Fuel Management

Nuclear fuel management deals with all activities important for an economical and safe utilization of nuclear fuel in a nuclear facility. In the past, most of the work in nuclear fuel management has been done for LWRs, the most commonly deployed power reactors in the world. While the fuel management procedures and methodologies are well established for PWRs and BWRs, far less information is available on the nuclear fuel management of other reactor types like VHTRs or liquid metal cooled fast reactors. Nevertheless, the general fuel management principles described in the following apply for all reactor types with a batch-wise refueling regardless.

In this section, all the most important aspects of nuclear fuel management are presented, including safety and fuel economy considerations; the difference between out-of-core and in-core nuclear fuel management is explained and an

overview of nuclear fuel management problems and solution methods is given, followed by a brief description of common nuclear fuel management reactor core analysis. Special attention is given to the nuclear fuel management of liquid metal cooled fast reactors and research reactors.

2.1.1 Scope and Aims of Nuclear Fuel Management

According to Graves (1979), "Nuclear fuel management encompasses activities involved in procuring, designing, fabricating, and irradiating nuclear fuel, as well as the decisions in shipping, spent fuel reprocessing, and in radioactive waste disposing." In this work, however, nuclear fuel management is perceived rather as a complex multidisciplinary decision making process that directly influences reactor core properties such as its reactivity and neutron flux, power, and burnup distributions. This more traditional formulation limits the scope of nuclear fuel management to nuclear fuel design and (re-)use of the nuclear fuel in the reactor core. It also assumes that the reactor core geometry as well as the fuel assembly mechanical design and, hence, its thermal-hydraulics profile, are fixed and cannot be influenced by the nuclear fuel management decisions.

Aims

The general objective of nuclear fuel management may be defined as an effective use of nuclear fuel in a nuclear reactor that meets a set of demanded performance criteria in a safe and most efficient way. The nature of the set of performance criteria usually depends on the purpose of the nuclear facility; for instance, power reactors operated by commercial utilities aim at demanded production of electric energy over a defined period of time, while material testing reactors operated by research laboratories aim at achieving required irradiation conditions during the cycle expressed by the accumulated neutron fluence, for example[1]. The general requirement on the process efficiency is typically expressed in terms of some cost-based parameters to be optimized, resulting in a core reload design with a minimum fuel cycle[2] cost in dollars.

[1]The target quantities like the electric energy produced or the neutron fluence will be collectively called quantities of interest (QOIs). Another hypothetical example of QOI could be the mass of fission materials produced in a breeder reactor or the amount of MAs transmuted in a burner reactor.

[2]The term fuel cycle used here refers to the irradiation process that begins at the time of reactor startup after refueling and ends when the reactor is shut down for subsequent refueling. This use of fuel cycle is in contrast to the broader use of the term to describe all the processes involving fuel, from mining to fabrication, irradiation, reprocessing, and storage (Tsvetkov et al., 2012). The first, more specific meaning is assumed in the remainder of the text unless stated otherwise.

Safety Considerations

The first and for this work also the most relevant objective of nuclear safety can be formulated as "to protect public health by reducing the risk from releases of radioactivity to acceptable levels," (Sackett, 2012). This is accomplished by providing a design that assures safe, stable, and reliable operation by preserving key safety functions, that is, containment of radioactive material, reactor shutdown, and residual heat removal. The key safety functions are achieved through the application of the defense-in-depth approach in safety design that employs basic principles of diversity, redundancy, and independence (IAEA, 1996). In order to demonstrate compliance with regulatory requirements, the proposed design has to provide large safety margins and additional safety features that allow normal operation without challenges to safety limits and protect against the unlikely faults foreseen within the safety design basis. The most demanding events, also identified as design basis accidents (DBAs), are analyzed and the results are documented in safety reports to verify safety margins for licensing.

The same general safety analysis approach is applied for thermal and fast reactors cooled by liquid metals. However, the different design features of LWRs and fast reactors imply different problems in the area of reactor transient and accident analysis and give both technologies unique safety characteristics.

Safety related advantages and disadvantages of SFRs (that can be related to any other fast reactor designs) in comparison with thermal reactors were summarized by Sackett (2012). The main SFR characteristics that make achieving exceptional safety levels more challenging than in the case of LWRs are higher core power density, shorter neutron lifetime, lower effective delayed neutron fraction, positive sodium void effect, less negative Doppler reactivity coefficient, rather violent sodium interaction with air or water, and the fact that the reactor core is not arranged in its most reactive configuration. The favorable safety characteristics of SFRs are sufficiently high boiling point in sodium that makes it possible to operate the reactor near atmospheric conditions, very high heat capacity and thermal conductivity, negligible spatial power shifts, and the absence of the xenon poisoning issue.

Nuclear fuel management decisions must comply with safety standards and operational limits in order to prevent from nuclear and radiation accidents or to limit their consequences. A safety analysis has to be completed for each core loading to show that "at no time during normal operation and anticipated transients should the fuel be exposed to conditions which may cause damage," (S. H. Levine, 1986).

The plant accident and transient analysis is computationally demanding. In

order to simplify the whole process, the so-called envelope approach is usually adopted that says that the plant safety is guaranteed if the so-called envelope criteria established based on the transient and accident analysis of the initial (envelope) core are not exceeded. Hence, only a limited number of reactor core characteristics has to be supplied to prove the reactor core safety. The reactor core reload is accepted if its characteristics meet the envelope criteria, otherwise, a complete set of safety calculations has to be performed to prove or disprove its acceptability.

Two Subproblems

Since nuclear fuel resides in a reactor core for several cycles, a multi-cycle assessment is required to make nuclear fuel management decisions that would ensure the best use of it in the long-term. It allows to prevent from the "rob Peter to pay Paul" effect, when the decisions optimized for an upcoming reload cycle may penalize subsequent cycles, leading to a less favorable net fuel economy. The multi-cycle analysis is usually based on projected estimates of fuel cycle lengths, per-cycle quantity-of-interest (QOI) levels (see Footnote 2.1.1 on page 12), and/or other operational requirements supplied by the utility roughly one year before the beginning of the upcoming cycle. Target margins to limits must be large enough to accommodate expected changes in the utilization plan, malfunctions, etc.

In order to license a proposed core, a single-cycle analysis is usually performed several months prior to the reactor startup, reflecting all changes in the design assumptions. At this stage, the fuel cycle economy can be improved by optimizing the core reload design within the boundary conditions provided by the results of the multi-cycle analysis.

The multi-cycle analysis usually falls within the scope of so-called out-of-core nuclear fuel management and the single cycle problem is usually addressed by so-called in-core nuclear fuel management. Even though this subdivision into two tightly coupled subproblems is purely artificial, it is a commonly used practice in nuclear fuel management. Since the objective of this dissertation belongs to in-core fuel management, it is described in a greater detail in the following section.

2.1.2 In-Core Nuclear Fuel Management

With the out-of-core decisions made on the number and composition of fresh fuel assemblies and shuffled fuel assemblies to load to the reactor, in-core (or

single cycle) nuclear fuel management determines the core design and reactor operation variables for an upcoming cycle. This phase also requires relatively detailed evaluation of the nuclear, thermal, and hydraulic characteristics of the plant to assure that the technical specifications and other constraints imposed by the operating facility are met (Graves, 1979).

Objectives

The objective of in-core nuclear fuel management is consistent with the minimum fuel cycle cost objective of out-of-core nuclear fuel management. However, the levelized fuel cycle cost (LFCC), which determines the fuel cycle over a multi-cycle planning horizon, can no longer be calculated for a single cycle. Therefore, other objective functions are used in single-cycle optimization that attain the same goal with a greater mathematical simplicity.

Most often, non-cost surrogate objectives are utilized to minimize the cost of fresh assemblies loaded into the reactor at the beginning of cycle such as to minimize the average feed fuel enrichment per unit of QOI produced during the cycle or to maximize cycle QOI production for the fixed cycle length. Equivalent alternatives include maximizing the end of cycle (EOC) fissile inventory, maximizing the beginning of cycle (BOC) or EOC k_{eff}, or maximizing the fuel cycle length. In some cases, minimizing the number of feed fuel assemblies may also be considered. Another example of an in-core fuel management objective that can lead to some fuel cost savings is the maximization of the discharge fuel burnup.

Finally, specific objective functions can be used for particular nuclear reactor types like PWRs or BWRs, for which it may be desirable to maximize the EOC soluble boron concentration or to minimize the EOC core flow rate, respectively. In the recent paper by Kalcheva and Koonen (2012), the BR2 core configuration was optimized to enhance nuclear safety and minimize the fuel cycle cost (the amount of fissioned ^{235}U) for the same or increased cycle length at a given reactor power level.

Decision Variables

The in-core fuel management decisions address the arrangement of fuel assemblies in a reactor core along with the excess reactivity control strategy. The arrangement of the fuel in the reactor is determined by the loading pattern, which specifies the location (and orientation) of the fresh and partially depleted fuel. The location of integral burnable poisons within the core is also described

by the loading pattern, because the placement of the burnable poisons is a part of the lattice assembly. The number of the feed fresh assemblies typically obtained from the multi-cycle analysis may also be one of the in-core variables. For BWRs, additional decisions enter in-core fuel management concerning the control rod (blades) pattern design, the Control Rod Programming (CRP) problem, and the core flow rate plan.

Constraints

In-core nuclear fuel management constraints include constraints on the fuel cycle length and QOI production implied by the cycle operational requirements, the discharge burnup limits (fuel pellet, rod, bundle, and batch) specified by a fuel vendor to guarantee the fuel performance and limit its damage, the reactivity and thermal limits to ensure safe operation throughout the entire cycle, and the loading pattern design rules.

The loading pattern design rules are important mainly for formal optimization techniques. They include simple allocation rules like one assembly per position rule, exclusion and inclusion rules, that is, constraints on the position and orientation of certain assemblies in the core, and constraints imposed by the inventory of available fuel.

As was just outlined, many objective and constraint functions can be considered in in-core nuclear fuel management. In real situations though, only those objectives and constraints are considered that suit best the type, purpose, and needs of a particular facility. Turinsky (2010) mentions constraints usually used in the fast reactor in-core fuel management analysis: the peak linear power density, the maximum cladding fluence, the maximum discharge burnup, and the limits on various reactivity coefficients and reactivity device worths.

Fuel Loading Patterns

The large number of variables and constraints makes the in-core nuclear fuel management a difficult problem to deal with every time a reactor is reloaded (see Section 2.2). Therefore, several loading patterns have been developed in the past by nuclear analysts based on their expert knowledge and intuition with the goal to maximize the cycle QOI production and to minimize power peaking, or in other words, to increase the margins to thermal limits. Those loading patterns, which have been widely applied in the nuclear industry, are described below. They represent the most common solutions to the core loading problem.

Out-in pattern. The so-called out-in checker-board loading pattern was for many years the preferred loading pattern. In the checker-board loading pattern, the fresh fuel assemblies are loaded on the core periphery while the remaining fuel assemblies are arranged in the central zone. This fuel arrangement was favored for its uniform (flat) power density distribution across the core. However, having the fresh fuel on the core outskirts increases neutron leakage, which in turn results in less energy being extracted from the core and increased neutron-induced damage to internals and reactor vessel, worsening the fuel economy (Turinsky, 2010).

In-out and low-leakage patterns. An alternative fuel reloading strategy has been developed that maximizes cycle energy production by minimizing neutron leakage and by shifting power from regions of lower neutron importance to regions of high neutron importance. In the in-out core arrangement, the reduced neutron leakage is achieved by placing a specified fraction of partially burned fuel assemblies on the core periphery and moving fresh fuel towards the core center. Besides the improved fuel cycle cost, this reloading strategy brings with it the additional advantage of reducing pressure vessel irradiation levels and, as a drawback, some loss in margin to the peak-to-average power density limits (Graves, 1979).

Zonal refueling. The above described fuel loading strategies are characteristic mainly for LWRs. Limited information is available on the fuel loading strategies of fast reactors. Turinsky (2010) describes a typical fast reactor in-core fuel management scheme as the one consisting of two or three radial zones of different fuel reactivities achieved by elemental mixtures, all done to flatten the power distribution. In this arrangement, fresh fuel is loaded into one of these zones, replacing irradiated fuel that moves to another zone and so on, until the fuel in the last zone is discharged.

Scatter pattern. The long mean free path of neutrons in fast reactors, makes the utilization of a no-fuel shuffling scheme more plausible, where a fraction of the fuel in each radial zone may be replaced with fresh fuel assemblies without movement of any partially burnt assemblies (Turinsky, 2010). The advantage of the scatter loading[3] is that it does not require the time-consuming step of shuffling fuel in each refueling outage and limits the undesirable manipulation with the fuel assemblies in an opaque environment.

[3]Another name used for the no-fuel shuffling scheme; see Tsvetkov et al., 2012 or S. H. Levine, 1986, for instance.

Optimization Problems

Given various out-of-core and in-core nuclear fuel management goals, decision variables, and constraints, nuclear fuel management constitutes various optimization problems. The fuel assembly design problem is solved for BWRs to find an optimum axial fuel-assembly fuel profile. The so-called CRP is another problem solved for BWRs. The goal here is to optimize the control blade insertion patterns as a function of cycle exposure. The problem typically solved for both BWRs and PWRs is the fuel lattice design problem, which aims at an optimum arrangement of different fuel pins within a fuel assembly. More than with the power distribution within a fuel assembly, however, a fuel manager is concerned with power sharing between the fuel assemblies determined by the fuel loading pattern. The arrangement of the fuel assemblies in the reactor core has major influence on the radial power distribution of the core, whereas the fuel lattice and assembly design problems are of a secondary importance. As such, the loading pattern optimization problem has been the most studied in-core fuel management optimization problem. It is described in detail in Section 2.2.

The fuel assembly, fuel lattice, loading pattern, and control rod pattern design problems are all tightly coupled together and should be solved as one nuclear fuel management problem in the ideal case. However, the resulting problem would be simply too complex to be handled (large number of variables), and therefore different problems have to be solved separately for different reactor types. Only the loading pattern optimization problem is considered in the fast reactor fuel management analysis.

The same underlying phenomena—neutron transport and fuel depletion (see Section 2.3.1)—and the same nature of the optimization objectives, constraints, and variables, makes it possible to use the same techniques for solving all nuclear fuel management optimization problems at hand. An overview of the techniques applied specifically to the loading pattern optimization problem is given in Section 2.2.4.

2.1.3 Analytical Practices

The process of designing optimum core reload involves performing several nuclear fuel management calculations diverse in their goals and employed methods. The same procedures are used in this process to evaluate the core neutronics and thermal-hydraulics properties of both LWRs and fast reactors, although the range of parameters and problem emphases are different (Graves, 1979).

Distinct analytical tools are used at the out-of-core and in-core fuel management

levels, ranging from simple scoping models used to evaluate long term fuel cycle strategies (e.g., to set cycle length or to analyze reactor economics), through preliminary design methods used for loading pattern optimization, to high-fidelity multi-physics codes used for licensing analysis. The whole analytical sequence leading to a new reactor core reload design can be represented by a simple scheme shown in Figure 2.1. Analytical methods used in different stages of the sequence are described below in this section.

Scoping Methods

The sequence begins with setting initial conditions for a multi-cycle scoping analysis. These conditions may include updated information on projected burnup of the fuel assemblies to be potentially reused in the new core loading, reactor operation scheduling, QOI requirements, etc. The actual multi-cycle scoping analysis then takes place that provides the optimum solution to the out-of-core fuel management problem; the mutually dependent fuel cycle parameters like the feed fuel enrichment, the discharged fuel burnup, the fuel batch size, or the cycle length are selected that meet the constraints on the maximum feed fuel enrichment, the discharged fuel burnup, and others, while minimizing the LFCC. This is usually achieved by using simplified point reactor models, in which all spatial detail below that of the batch-average level is suppressed and the principal focus is on the fuel depletion (Downar and Sesonske, 1988; Y.-S. Park et al., 1994). An example of a successful and well established point reactor model is the linear reactivity model (Driscoll et al., 1990).

The results of the scoping analysis are used as an input to the next phase, in which a preliminary reactor core reload design is specified by solving the in-core fuel management problem.

Preliminary Design Methods

The primary purpose of the preliminary design methods is to find an optimum refined candidate reactor core design (i.e., a loading pattern and a CRP for BWRs) that complies with the out-of-core decisions made in the previous step, and to calculate its main characteristics that are necessary for making conclusions on its acceptability by comparing them with the in-core nuclear fuel management constraints mentioned in Section 2.1.2.

Two essential characteristics to be calculated are the core reactivity and the power density distribution in the core. The methods should be preferably fast enough so they can be utilized by modern optimization techniques, which require

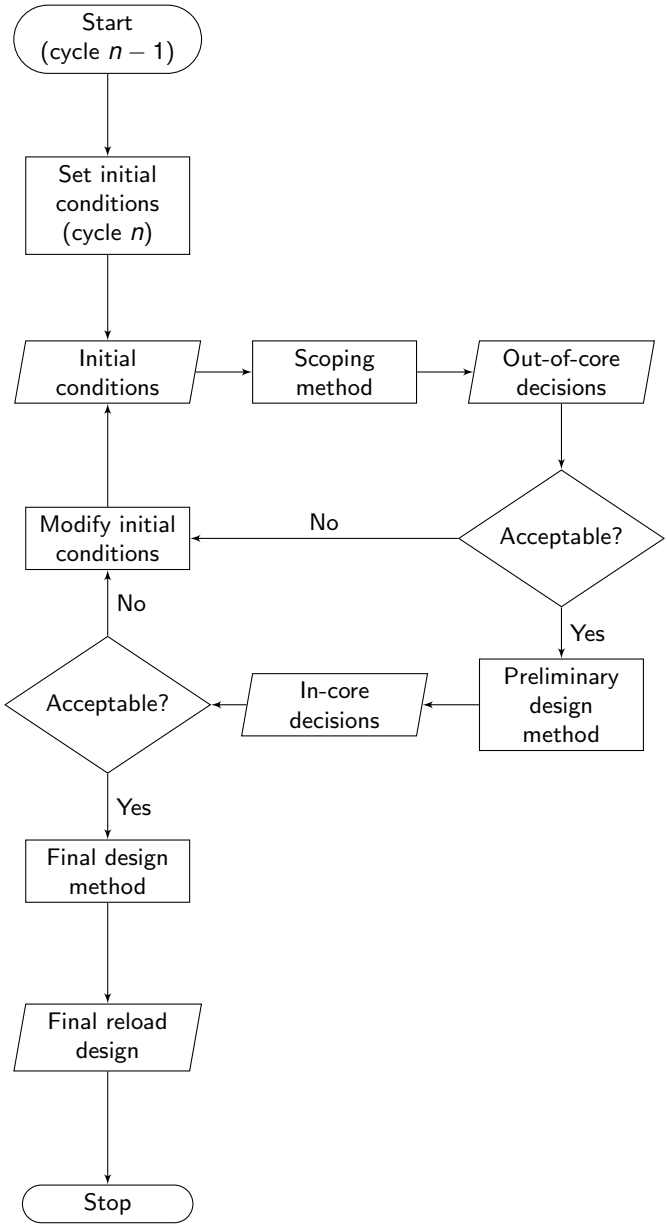


Figure 2.1: Analytical sequence for reactor core reload design

evaluation of numerous loading patterns and CRP designs (Section 2.2). The preliminary design methods are hence typically represented by two- or, in case of BWRs, three-dimensional coarse-mesh multi-group nodal diffusion methods, in which the spatial detail is limited to the fuel assembly and the energy detail is limited to two energy groups in case of LWRs and more energy groups in case of fast reactors. Since the acceptability of a preliminary core reload design frequently depends on the maximum fuel pin power density, pin-by-pin solutions are constructed by dehomogenization of the nodal results using heterogeneous form functions for power burnup and fluxes (IAEA, 1995).

Core reactivities and power distributions of advanced reload configurations containing burnable poisons need to be calculated at various burnup steps to make sure that the design constraints are met throughout the cycle. The reload analysis is simplified in absence of burnable poisons, in which case the configurations need only to be compared at the beginning of the cycle, when the most restrictive conditions with respect to power peaking occur (Driscoll et al., 1990).

Once the preliminary core configuration is satisfactory and meets all safety related constraints imposed at this stage of the reload design process, final design methods are used for its licensing. Otherwise, the original initial conditions have to be relaxed and new scoping and preliminary design analyses have to be performed repeatedly until an acceptable fuel loading design is found.

The reactor analysis codes used in this work for modeling MYRRHA (Chapter 3) fall into the category of preliminary core reload design methods.

Final Design Methods

Final design methods employed for core design licensing analysis usually include detailed multi-group fine-mesh methods with a spatial detail on the pin-by-pin level. Tens of calculations are performed at this stage to elaborate the fundamental physics and safety characteristics of the concerned cycle. This includes determination of core attributes over a normal operating range (e.g., power and burnup distribution, reactivity coefficients, control rod worths, point kinetic parameters), determination of core attributes for safety analysis done by point kinetics, and three-dimensional core simulation within or coupled to a system transient code for analysis of selected accidents (Turinsky, 2010). Additional calculations are needed for startup physics tests and for the actual reactor operation.

2.2 Loading Pattern Optimization Problem

As explained in Section 2.1.2, the loading pattern optimization problem (LPOP)[4] is an important nuclear engineering problem reactor core designers have to solve in the preparation of a new core reload. Since the problem of finding an optimum loading pattern is the central research topic of this work, it will be described in a greater detail hereinafter.

2.2.1 Definition

LPOP can be simply formulated in words as follows: Find an optimum LP with respect to a given objective, such that it satisfies all given constraints. A more formal definition of the problem requires introduction of the term objective function.

Objective Function

The objective function $f(\boldsymbol{x})$ is a mathematical function of n decision variables $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ that is to be optimized. The objective function[5] is a function that assigns to each feasible loading pattern a certain value as a measure for fulfilling the objective(s):

$$f(\boldsymbol{x}) : \mathcal{F} \longrightarrow \mathbf{R} \ . \tag{2.1}$$

$\mathcal{F}$ is the set of the feasible solutions in an n-dimensional space and $\mathbf{R}$ is the set of real numbers.

The optimum solution $\boldsymbol{x}_{\mathrm{opt}}$ can thus be defined as the solution with the largest or lowest achievable objective function value $f(\boldsymbol{x}_{\mathrm{opt}})$, depending on whether a maximization or minimization problem is considered.

Objectives and Constraints

LPOP objectives and constraints were discussed in detail in Section 2.1.2. To recall the most important objectives one should mention the maximum cycle

[4] Alternative names can be found in the literature for the LPOP such as the in-core fuel management optimization problem (ICFMOP), the nuclear reactor reload problem (NRRP), the nuclear reactor reload core design problem (NRRCDP), and others.

[5] Also known as the criterion function, performance measure, performance index, or fitness function (see Section 2.2.4). Other names for an objective function are a cost function when it is minimized and a revenue function when it is maximized.

QOI production, the minimum BOC power peaking factor, or the maximum BOC or EOC k_{eff}. Among the constraints, the power peaking factor P_{q} plays the key role. It is defined as the ratio of the peak to average power densities in the reactor core:

$$P_{\text{q}} = P_{\text{r}} \times P_{\text{z}} \ , \tag{2.2}$$

where P_{r} and P_{z} are the radial and the axial power peaking factors, respectively. The loading pattern optimization has a direct influence on P_{r}, which is a product of two components, the whole-core per-assembly component $P_{\text{r core}}$ and the within-assembly component $P_{\text{r FA}}$:

$$P_{\text{r}} = P_{\text{r core}} \times P_{\text{r FA}} \ . \tag{2.3}$$

Also important are the fuel assembly allocation constraints. For example, assume a loading pattern with N_{FA} fuel assembly locations and an equal number of available fuel assemblies of n different ages ($n \leq N_{\text{FA}}$):

$$\sum_{i=1}^{n} l_i = N_{\text{FA}} \ , \tag{2.4}$$

where l_i is the number of the fuel assemblies of age i. Let a set of binary variables

$$x_{i,j}, \quad i = 1, 2, \ldots, N_{\text{FA}}, \quad j = 1, 2, \ldots, n$$

be defined so that

$$x_{i,j} = \begin{cases} 1 & \text{if the location } i \text{ will contain an assembly of age } j, \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

Then the fact that all fuel assemblies are loaded in the reactor core exactly once and that each loading pattern location contains exactly one fuel assembly can be described in the set of equations:

$$\sum_{i=1}^{N_{\text{FA}}} x_{i,j} = l_j, \quad j = 1, 2, \ldots, n \ , \tag{2.6}$$

$$\sum_{j=1}^{n} x_{i,j} = 1, \quad i = 1, 2, \ldots, N_{\text{FA}} \ . \tag{2.7}$$

General Definition

In its general form, the loading pattern optimization problem may be stated as a mixed integer nonlinear program (MINLP):

$$\underset{\boldsymbol{x}}{\text{Minimize}} \quad f(\boldsymbol{x}) \ , \tag{2.8}$$

$$\text{subject to} \quad g_j(\boldsymbol{x}) \leq 0, \qquad j = 1, 2, \ldots, m \ , \tag{2.9}$$

$$l_i \leq x_i \leq u_i, \qquad i = 1, 2, \ldots, n \ , \tag{2.10}$$

$$\boldsymbol{x} \in \mathbf{Z}^r \times \mathbf{R}^{n-r} \ , \tag{2.11}$$

where the objective function $f(\boldsymbol{x}) : \mathbf{R}^n \to \mathbf{R}$ and the constraints $g_j(\boldsymbol{x}) : \mathbf{R}^n \to \mathbf{R}$ are in general multivariate non-convex functions, m is the number of constraints, n is the number of variables, r is the number of integer variables, and $\boldsymbol{x}$ is the n-vector of variables, whose i-th component is denoted by x_i. l_i and u_i are lower and upper bounds on variable x_i.

The formulation of the loading pattern optimization problem as an MINLP involves a full description of the mathematical model of the problem or, in other words, a full description of the underlying physics phenomena by means of a closed set of algebraic equations, which is, in this case, the behavior of neutrons in a reactor core described by the transport equation (see Eq. (2.15) on page 34).

The loading pattern optimization problem as such belongs to a group of specific optimization problems treated by a number of specific optimization methods. In the next part, the classification of the loading pattern optimization problem within the domain of general optimization problems together with some explanations will be adduced.

2.2.2 Classification

Following Sarker and Newton (2008), each optimization problem can be classified with respect to several criteria like the type of optimization, the number of objectives, the presence of constraints and their properties, the type of the variables and the properties of the functions from the mathematical model, plus some others.

It is generally admitted that the loading pattern optimization problem is stated

as a minimization problem[6]. A particular problem may be a single or a multiple objective optimization problem depending on the number of objectives. The same can be said with regard to constraints; a problem is called constrained if it contains one or more constraints and unconstrained otherwise. For practical reasons, all LPOP constraints are usually expressed in an inequality form from Eq. (2.9). This is always possible also for equality constraints $h_j(\boldsymbol{x})$ by introducing a small enough positive real number (tolerance) $\epsilon \in \mathbf{R}^+$ and reformulating them into the inequality form:

$$g_j(\boldsymbol{x}) = \left|h_j(\boldsymbol{x})\right| - \epsilon \leq 0 \ . \tag{2.12}$$

Constraints that must be satisfied are known as hard constraints. Soft constraints, on the other hand, are constraints that can be violated with a certain penalty.

Depending on the nature of the problem, the variables in the model may be real or integer or a mix of both. Accordingly, continuous, integer, discrete, or mixed optimization problems are recognized. Although the loading pattern optimization problem is stated herein as a mixed problem, in most cases it does not include real variables and one deals only with integer or discrete variables, $n = r$ in Eq. (2.11)[7]. An optimization problem is then termed a combinatorial problem.

The function classification mainly deals with the functions' mathematical properties, which are very important from the solution approach point of view (Sarker and Newton, 2008). An objective or a constraint may be linear, nonlinear, or both. If all functions in the model are linear, it is called a linear optimization problem or a linear programming optimization problem. Otherwise it is called a nonlinear optimization problem. Analogously, one can classify optimization problems regarding the convexity of the functions—convex or non-convex[8] (concave), or their differentiability—differentiable or non-differentiable. Finally, a problem is called unimodal if the functions have only one peak (optimum solution), whereas it is called multimodal if they have more than one peak (either local or global optima).

Considering its nature, the loading pattern optimization problem is an example of a nonlinear non-convex optimization problem characterized by a lack

[6]Note that each maximization problem $\max_{\boldsymbol{x}} f(\boldsymbol{x})$ can be transformed into a minimization problem $\min_{\boldsymbol{x}} f(\boldsymbol{x})^{-1}$.

[7]An example of a variable $x_i \in \mathbf{R}$ that can be considered in an LPOP is the feed fuel enrichment or the burnable poison (BP) loading.

[8]Note that the fact that the model functions are non-convex does not imply non-convexity of the whole space of the decision variables. Nothing can be said about the convexity of the search space in this case (Leyffer, 2013).

of derivative information. Other important attributes are its very high computational complexity[9] and exceptionally large search space size.

2.2.3 Search Space Size

The search space size is estimated as the combinatorial number of all possible solutions to the problem. For example, assume the MYRRHA core loading with 69 fuel assemblies and a five-batch reloading scheme, meaning that approximately one fifth of the fuel assemblies is replaced at each refueling. The remaining fuel assemblies, having resided in different regions of the core for one to four cycles, are generally different. Hence, the fuel problem alone, assuming 15 identical fresh fuel assemblies loaded in the core, has $54! \times C(15, 69) \approx 1.3 \times 10^{86}$ possible solutions, since there are $C(15, 69)$ ways of choosing where to put the fresh fuel assemblies and 54! ways of arranging the others in the remaining positions.

The search space is usually reduced by assuming a rotational or reflective core symmetry. Should the one-third core rotational symmetry be considered, the number of possible solutions would decrease to $18! \times C(5, 23) \approx 2.2 \times 10^{20}$. Moreover, when the fuel assemblies are assumed to be identical in each of the fuel batches, the number of possible configurations will become smaller. For the reloading scheme $\boldsymbol{l} = (5, 5, 5, 4, 4)$, this number would equal to $\prod_{i=1}^{5} k_i \approx 2.6 \times 10^{13}$, since the $l_1 = 5$ fuel assemblies from the first batch can be loaded into the core in $k_1 = C(5, 23)$ different ways, the $l_2 = 5$ fuel assemblies from the second batch in $k_2 = C(5, 23 - 5)$ different ways, and so on. The size of the search space could be further decreased if there is a constraint, which says that some of the fuel assemblies must stay fixed in certain positions or may stay only in some positions due to operational or other reasons.

The search space size is much larger in case of LWRs, which have significantly more fuel assemblies (up to 800 for BWRs). Also, when working with a rotational quarter-core symmetry, for instance, each fuel assembly can be considered to have four possible orientations. In some cases, fuel assemblies can be put into a reactor even flipped upside down (Kalcheva and Koonen, 2012). The problem will be even more difficult if a new loading pattern will be designed assuming not only partially burned fuel assemblies present in the core but also those placed in the fuel storage pool. Finally, a batch of fresh fuel assemblies inserted into the core during reshuffling may consists of several different types of fuel assemblies with various enrichments and burnable poison loadings, which again makes the loading pattern optimization problem more challenging.

[9] LPOP belongs to the class of NP-hard problems.

2.2.4 Optimization Methods

The classification, the complexity, and the size of the loading pattern optimization problem make it a difficult optimization problem to solve. Concerning the very simple example from the preceding section and assuming that one loading pattern evaluation could be evaluated in one second of computer time, a direct exhaustive search for the best reload pattern would take nearly one million years. Hence, many sophisticated optimization methods have been used in the past that could solve the problem in a reasonable time, some of them being more and some of them being less successful. A list of the methods that have been developed and applied for solving the core reload problem is in Table A.1 in Appendix A. Even a short review of each of these methods would go far beyond the scope of the dissertation[10]. Therefore, only a brief description of two general optimization approaches distinct from the point of view of the used methodology will be given in the following. Two particular optimization methods applied to the MYRRHA loading pattern optimization problems will be described in detail in Chapter 4.

Most of the research has been done to solve fuel shuffling problems for power LWRs and only a few studies have been dedicated to other reactor types (cf. Table A.1). A limited amount of work has been published on refueling optimization for fast reactors (Kato et al., 1999; Kobayashi et al., 1976, 1977; Shvedov and Goncharov, 1992; Ziver, Pain, et al., 2004), ADSs (Ishida and Sekimoto, 2010a,b), or for the category of research reactors (Abd Elmoatty et al., 2011; Do and Nguyen, 2007; Keyvani et al., 2010; Mahlers, 1997; Schlünz et al., 2013; Shaukat et al., 2010; van Geemert, Quist, J. E. Hoogenboom, et al., 1998) or material testing reactors (Iqbal et al., 2008).

According to the author of this dissertation, it is reasonable to distinguish between two groups of optimization methods applicable to the loading pattern optimization problem[11]: methods based on a mathematical model of the problem and methods that use reactor analysis codes.

Methods using Mathematical Models

The first group of methods is represented by classical mathematical programming optimization methods that need a simple enough but reasonably accurate and case specific mathematical model of the optimization problem at hand. However,

[10] Brief historical reviews of loading pattern optimization studies can be found in Downar and Sesonske, 1988; S. H. Levine, 1995; Maldonado, 2005; Turinsky, 2005.

[11] The same optimization approaches and optimization methods can be used with some modifications for solving other out-of-core and in-core fuel management problems.

development of such a model is very challenging in case of the loading pattern optimization problem and requires use of many simplifying assumptions.

In order to describe the system by a tractable closed set of algebraic equations with a limited set of key decision variables, it is necessary to use a very approximate representation of the core physics and fuel cycle model. Commonly used simplifications may include discretization of continuous variables, linearization of equations, restriction to the equilibrium cycle, and others.

Several standard mathematical programming techniques have been applied for solving the core refueling problem. Among them Linear Programming (LP) (Kubokawa and Kiyose, 1975; Mahlers, 1997; Okafor and Aldemir, 1988; Sauar, 1971; Suzuki and Kiyose, 1971a) or Mixed Integer Nonlinear Programming (MINLP) (T. K. Kim and C. H. Kim, 1996; T. K. Kim and C. H. Kim, 1997; Klerk et al., 1997; Quist, van Geemert, J. E. Hoogenboom, Illés, C. Roos, et al., 1999; Quist, van Geemert, J. E. Hoogenboom, Illés, Klerk, et al., 1999; Quist, K. Roos, et al., 2001). A particularly interesing approach was taken by Allaire and Castro, who used the homogenization theory to relax the discrete optimization problem into a contiuous and well-posed problem, solved it using classical methods of optimal control, and recovered a discrete admissible distribution of assemblies by a numerical penalization technique. The main advantage of this method is that the resulting reload pattern is guaranteed to be near the optimum (Allaire and Castro, 2002).

The advantage of the model-based methods is a relatively small computational cost[12] and the fact that the optimum solution can be theoretically guaranteed in some special cases[13]. The problem frequently encountered with some mathematical programming methods in solving multimodal problems is the inability to distinguish local minima from a global optimum[14]. For instance, it is known that MINLP algorithms are not guaranteed to converge to a global optimum of a non-convex problem (Leyffer, 2013). Another drawback is that the preparation of the input parameters for the simplified model requires an expensive preprocessing step. These parameters also introduce additional discrepancies into the model that are reflected in the final results. However, the principal deficiency of the mathematical modeling approach, when it comes to solving the loading pattern optimization problem, lies in the simplifying assumptions in the mathematical model, since an optimum solution can be

[12]However, this is not true for modern mixed-integer nonlinear solvers, which are much less computationally efficient than those for LP and require considerable computer resources.

[13]These are often academic problems that have no practical use.

[14]Galperin (1995) investigated the landscape of the LPOP search space by analyzing 300 000 out of $\approx 10^{12}$ problem states. He revealed quite a complicated search space structure with an extremely large number of local peaks: about one peak per hundred configurations.

"generally only as good as the accuracy of the core neutronics model used to derive the solution," (Downar and Sesonske, 1988).

One way to overcome this issue is to narrow down the number of considered loading patterns using some kind of systematic procedures, evaluate these patterns with a more sophisticated reactor core analysis method, compare them, and select the best one.

Methods using Reactor Codes

The rapid development in the computer technology has allowed to consider new solution schemes for core reload optimization that are based on the examination of a larger number of core configurations by standard neutronics codes.

The use of neutronics codes instead of simplified mathematical programming models poses several advantages. The problem with additional data preparation is avoided and basically every reactor type and refueling strategy that can be evaluated by a reactor analysis code can be a subject of the loading pattern optimization, making this approach more flexible. More importantly, it allows to eliminate the need for significant simplifying assumptions characteristic for the previous approach, and thus, to deliver more precise results, the quality of which depends merely on the capabilities of the code of the choice. The use of the actual nonlinear system of equations and continuous variables, sensitivity to local parameter variations, and substantially increased accuracy have made the nuclear industry more enthusiastic about the potential of these methods.

Selecting a proper neutronics code is an important task in which one is looking for two mutually contradicting qualities: a maximum modeling precision and minimum computational requirements. The code systems used for licenseable design evaluation would be ideal candidates. However, they are so complex that it is not practical to incorporate them into the search procedure. Therefore, preliminary design methods are used rather than final design methods (Section 2.1.3), as they are sufficiently fast while providing an acceptable confidence in the principal results.

An alternative solution that many authors use for core loading optimization is based on surrogate models like artificial neural networks (Haykin, 1998); to name some of them: Erdoğan and Geçkinli, 2003; Fadaei and Setayeshi, 2008; Hedayat et al., 2009; Ortiz-Servin et al., 2011; Pazirandeh and Tayefi, 2012; Sadighi et al., 2002b; Ziver, Pain, et al., 2004. The application of such methods is case specific and involves a time consuming "learning" phase that counterbalances their favorable speed. The confidence in the results is rather low and typically confined to the solutions that stay within the boundaries

defined by the learning data sets. The use of artificial neural networks is usually avoided for these reasons.

The number of trials to be evaluated remains prohibitively large even if it should represent only a fraction of all feasible solutions and the most efficient reactor modeling methods should be used. This stimulates the need for very powerful optimization techniques so the number of trials needed to arrive to a high-quality solution could be reduced to an acceptable level (one talks typically about thousands to millions of trials).

Many techniques have been used to tackle the loading pattern optimization problem employing a full reactor core calculation, including deterministic neighborhood search procedures like the Direct Search (DS), also known as the hill-climbing method, or the Binary Exchange (BE) method (Y. J. Kim et al., 1987; Yamamoto, 1997). Other examples are expert or knowledge-based systems (Galperin, 1995; Galperin, Kimhi, et al., 1989; Galperin and Kimhy, 1991; Li, 1993; K.-J. Lin and C. Lin, 1998; Rothleder et al., 1988; Tahara et al., 1991). A particular group of methods that have received a lot of popularity since the late 1980s are so-called metaheuristic optimization methods or simply metaheuristics.

Metaheuristics. Metaheuristics are general purpose approximate search methods and well suited for highly-complex multimodal nonlinear combinatorial problems such as the loading pattern optimization problem. The term metaheuristic was first introduced by Glover (1986) and it derives from a composition of two Greek words: heuriskein (εὑρίσκω), which means "to find", and the prefix meta (μετά), which means "beyond, in an upper level" (Blum, 2005). According to Talbi (2009), metaheuristics may be viewed "as upper level general methodologies that can be used as a guiding strategy in designing underlying heuristics to solve specific optimization problems,"[15]. Indeed, these algorithms are "higher level" heuristics (in contrast with problem-specific heuristics), generally applied to problems for which there is no satisfactory problem-specific algorithm to solve them. As such, they have been widely used to solve complex problems in industry and services, in areas ranging from finance to production management and engineering (Boussaïd et al., 2013). Boussaïd et al. (2013) sum up characteristics of almost all metaheuristics: they are nature-inspired (based on some principles from physics, biology or ethology); they make use of stochastic components (involving random variables); they do not use the gradient or Hessian matrix of the objective function; they have several parameters that need to be fitted to the problem at hand.

[15] Before this term was widely adopted, metaheuristics were often called modern heuristics (Reeves, 1993).

Metaheuristics can be divided into two groups of methods, based on the common general optimization concept they share (Talbi, 2009). The first group are single solution-based metaheuristics, which improve a single solution by iterative search procedures that move (or "walk") from one solution to another through neighborhoods or search trajectories through the search space of the problem at hand. Examples of such methods are different local search methods, Simulated Annealing (SA) (Kirkpatrick et al., 1983), or Tabu Search (TS) (Glover and Laguna, 1997). The second group of methods are population-based metaheuristics, which rely on an optimization concept that can be described as an iterative improvement in a population of solutions; that is, one population of solutions is replaced by a new population of improved solutions repeatedly throughout the optimization process until a given condition is satisfied. The population-based metaheuristics include so-called Evolutionary Algorithms (EAs) like Genetic Algorithm (GA) (Goldberg, 1989; Holland, 1975), Evolution Strategy (ES) (Rechenberg, 1965), Evolutionary Programming (EP) (Fogel et al., 1966), and Genetic Programming (GP) (Koza, 1992), various swarm-intelligence techniques such as Particle Swarm Optimization (PSO) (Kennedy and Eberhart, 1995) or Ant Colony Optimization (ACO) (Dorigo and Stützle, 2004), and other methods.

The first applications of metaheuristics for solving the loading pattern optimization problem date back to the late 1980s (Kropaczek, 1989; Parks, 1987), while the number of research papers published on the topic has been growing steadily ever since then[16]. Examples of successfully adopted metaheuristic methods include EAs like GA (Alim, Yilmaz, et al., 2009; DeChaine, 1995; Kobayashi and Aiyoshi, 2003b; Parks, 1997; Pereira and Sacco, 2008; Poon and Parks, 1993; Yamamoto and Hashimoto, 2002; Ziver, Pain, et al., 2004) or Population-Based Incremental Learning (PBIL) (Baluja and Davies, 1998; Silva and Schirru, 2010; Silva and Schirru, 2014), Swarm Intelligence (SI) methods, including PSO (Abbassi et al., 2012; Babazadeh et al., 2009; Jamalipour, Sayareh, et al., 2013; Khoshahval, H. Minuchehr, et al., 2011; Meneses and Schirru, 2006), ACO (De Lima, Schirru, et al., 2008; Esquivel-Estrada et al., 2011; Hoareau, 2008; C. Lin and B.-F. Lin, 2012; L. Machado and Schirru, 2002; Silva, Schirru, and Lima, 2011; Wang and C. Lin, 2009), Artificial Bee Colony (ABC) (I. M. S. d. Oliveira and Schirru, 2011; Safarzadeh, Zolfaghari, Zangian, et al., 2014), or Firefly Algorithm (FA) (Poursalehi, Zolfaghari, and A. Minuchehr, 2013b; Poursalehi, Zolfaghari, A. Minuchehr, and H. K. Moghaddam, 2013), and some others like SA (Fadaei, Setayeshi, and Kia, 2009; Kropaczek and Turinsky, 1991; H. C. Lee et al., 2001; Mahlers, 1995; Parks and Suppapitnarm, 1999; Stevens et al., 1995; Yamamoto and Hashimoto, 2000), TS (Ben Hmaida et al., 1999; Castillo, Alonso, et al., 2004; Hill and Parks, 2015; Jagawa et al.,

[16]The author of this dissertation has recorder over 130 studies published on the subject between 1987 (one paper) and March 2014 (16 papers in 2013).

2001; C. Lin, J.-I. Yang, et al., 1998), or Harmony Search (HS) (Aghaie et al., 2013; Nazari et al., 2013; Poursalehi, Zolfaghari, and A. Minuchehr, 2013c; Schlünz et al., 2013). Note that only a few of the most recent references from the authors who have been most active in the field were cited here for each of the methods as their complete list would be too long (cf. Table A.1).

The principles of metaheuristic optimization methods will be illustrated on two examples in Chapter 4, where one well established (GA) and one more recent (ACO) metaheuristic method will be applied for solving MYRRHA loading pattern optimization problems.

As pointed out by Turinsky (1999), since the nuclear fuel management problem is highly constrained, leading to a disjoint feasible decision space, it is very important to appropriately treat constraints. For constraints that require assessment of the core attributes like the power peaking factors, if such constraints are treated as true-false, it is unlikely that the infeasible decision spaces will be transversed, preventing the determination of the vicinity of the true optimum. For this reason, proper implementations of the optimization methods bearing on an intelligent selection of a few candidate solutions utilize soft constraints.

Constraints Handling

A traditional constraint handling technique for treating soft constraints is based on objective function penalization. The objective function $f(\boldsymbol{x})$ to be minimized is deteriorated by adding to it a positive weighted term for each violated constraint $g_j(\boldsymbol{x})$:

$$f_{\mathrm{a}}(\boldsymbol{x}) = f(\boldsymbol{x}) + \sum_{j=1}^{m} \omega_j \, g_j(\boldsymbol{x})^{\alpha_j} \ . \tag{2.13}$$

$f_{\mathrm{a}}(\boldsymbol{x})$ is the augmented (extended) objective function and the weighting and scaling multipliers ω_j and α_j are called penalty factors. For ω_j:

$$\omega_j \begin{cases} > 0 & \text{if the constraint } g_j \text{ is violated}^{17}, \\ = 0 & \text{otherwise.} \end{cases} \tag{2.14}$$

The level of penalization should be proportional to the level of violation of the constraint and to its relative importance. The penalty factors ω_j can be general functions $\omega_j \equiv \omega_j(\boldsymbol{x})$.

[17] I.e., $g_j > 0$, cf. Eq. (2.9).

2.3 Nuclear Reactor Core Analysis

The goal of nuclear reactor core analysis is to evaluate reactor core characteristics needed for the solution of a broad range of reactor design and operational problems. In the following, the focus will be on nuclear fuel management and, in particular, on the analytical methods employed for preliminary design of a new fuel reload. Readers that are primarily interested in the optimization part of this dissertation, may safely skip this entire section.

The primary goal of the preliminary design methods (see Section 2.1.3) is to carry out a power distribution study of a new fuel reload design both to assess the performance of the new reload design and to establish that all the fuel in the reactor will operate at power densities well below conditions that would lead to failure.

The fundamental physics phenomena that govern the reactor core performance and safety characteristics, including the power distribution, are mentioned in the first part of this section. The computational routes[18] that make the established physics problems practically solvable and applicable in the preliminary design stage are then briefly described in the following part of the section. The last part is devoted to the specifics of fast-reactor and ADS physics.

2.3.1 Fundamental Problems

Nuclear reactor performance characteristics like the power or neutron flux distribution are generally determined by the neutron distribution in the reactor core, described by statistical models developed within the neutron transport theory. A steady-state neutron transport problem is typically solved by the preliminary design methods to estimate the distribution of neutrons in phase space (position and velocity vectors) under stationary conditions, whereas the effects of fuel depletion are treated separately.

The estimation of safety related parameters like the maximum fuel cladding or fuel center-line temperatures requires a detailed thermal-hydraulics analysis of the hot (sub)channels, which are the coolant (sub)channels with the highest linear power density or exit coolant enthalpy. In the preliminary design phase of a new core reload, however, the power peaking factors coming from the neutronics analysis are often used to preconceive whether the reload design violates the safety constraints or not, rather than the hot channel factors, which greatly

[18] The focus is only on deterministic computational schemes in this section.

simplifies the whole analysis[19]. The problem of the thermal-hydraulics (T-H) analysis of nuclear reactors itself constitutes a complete standalone Nuclear Engineering discipline, even a short description of which would go beyond the scope of this text. Readers interested in the topic are instead referred to the specialized literature; for example, Todreas and Kazimi, 1990a,b (general T-H problems of nuclear systems), or Tang et al., 1978 (T-H analysis of LMFBRs).

Neutron Transport

Transport equation. The transport of neutrons through bulk media is usually described by the differential form of the linear Boltzmann equation:

$$\begin{aligned}\frac{1}{v(E)}\frac{\partial}{\partial t}\,\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t) = &-\boldsymbol{\Omega}\cdot\nabla\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t)\\ &-\Sigma_{\mathrm{t}}(\boldsymbol{r},E,\boldsymbol{\Omega},t)\,\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t)\\ &+Q(\boldsymbol{r},E,\boldsymbol{\Omega},t)\ ,\end{aligned} \tag{2.15}$$

where

$\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t)$	is the angular neutron flux, a function of seven variables: spatial vector $\boldsymbol{r}\in\mathbf{R}^3$, neutron energy E, angular neutron vector $\boldsymbol{\Omega}\in\mathbf{R}^2$, and time t,
$v(E)$	is the neutron speed,
$\Sigma_{\mathrm{t}}(\boldsymbol{r},E,\boldsymbol{\Omega},t)$	is the macroscopic total cross section, and
$Q(\boldsymbol{r},E,\boldsymbol{\Omega},t)$	is the neutron source density.

The macroscopic cross sections from Eq. (2.15) are a priori functions of all phase space variables. However, their dependence on the neutron direction can be neglected for materials that are isotropic in space. Furthermore, for practical applications, the time variations in the cross sections are slow compared to the average neutron lifetime, which means that their effects on the neutron flux can be also neglected and treated as the separate fuel depletion problem described later in this section.

Regular nuclear fuel management calculations are done assuming a reactor operating in steady-state conditions. In this case, the Boltzmann equation reduces to

$$\boldsymbol{\Omega}\cdot\nabla\psi(\boldsymbol{r},E,\boldsymbol{\Omega})+\Sigma_{\mathrm{t}}(\boldsymbol{r},E)\,\psi(\boldsymbol{r},E,\boldsymbol{\Omega})=Q(\boldsymbol{r},E,\boldsymbol{\Omega})\ , \tag{2.16}$$

[19]This is true mainly for PWRs or FRs; in BWRs, the thermal hydraulics effects must be coupled with the neutronics analysis for all calculations including scoping and preliminary calculations (S. H. Levine, 1986).

with the steady-state source density term in the form

$$Q(\boldsymbol{r}, E, \boldsymbol{\Omega}) = \int_{4\pi} \int_0^\infty \Sigma_\mathrm{s}(\boldsymbol{r}, E \leftarrow E', \boldsymbol{\Omega} \leftarrow \boldsymbol{\Omega}')\, \psi(\boldsymbol{r}, E', \boldsymbol{\Omega}')\, \mathrm{d}E'\, \mathrm{d}^2\Omega'$$

$$+ \frac{1}{4\pi} \sum_{i=1}^{N_\mathrm{fiss}} \chi_i(E) \int_0^\infty \nu_i\, \Sigma_{\mathrm{f}\,i}(\boldsymbol{r}, E')\, \phi(\boldsymbol{r}, E')\, \mathrm{d}E'$$

$$+ \frac{1}{4\pi} q(\boldsymbol{r}, E)\ , \tag{2.17}$$

where

$\Sigma_\mathrm{s}(\boldsymbol{r}, E \leftarrow E', \boldsymbol{\Omega} \leftarrow \boldsymbol{\Omega}')$ is the macroscopic double differential scattering cross section that takes into account diffusion and (n,xn) reactions,

N_fiss is the total number of fissionable nuclides,

ν_i is the total number of neutrons produced per fission of the i-th fissionable nuclide,

$\Sigma_\mathrm{f}(\boldsymbol{r}, E)$ is the macroscopic fission cross section of the i-th fissionable nuclide,

$q(\boldsymbol{r}, E)$ is the isotropic external neutron source,

$\chi_i(E)$ is the neutron fission spectrum of the i-th fissionable nuclide, normalized to one,

$$\int_0^\infty \chi_i(E)\, \mathrm{d}E = 1, \quad i = 1, 2, \ldots, N_\mathrm{fiss}, \text{ and} \tag{2.18}$$

$\phi(\boldsymbol{r}, E)$ is the angle-integrated neutron flux defined as

$$\phi(\boldsymbol{r}, E) = \int_{4\pi} \psi(\boldsymbol{r}, E, \boldsymbol{\Omega})\, \mathrm{d}^2\Omega\ . \tag{2.19}$$

Introducing the transport (destruction) operator A, accounting for neutron leakage, absorption, and scattering transfer, and the fission (production) operator F, accounting for fission neutron production, the steady-state problem (2.16)–(2.17) can be expressed in a simplified form as

$$\mathrm{A}\psi = \mathrm{F}\psi + q\ . \tag{2.20}$$

The nature of the source terms in problem (2.20) implies two physically meaningful solutions of the problem. One of them is so-called source problem with $q \neq 0$, which can be solved only for subcritical systems, as there is no time-independent physical solution for critical and supercritical systems with sources.

Mathematical solutions would have negative fluxes in this case. However, since practically all nuclear reactors are critical systems without the external neutron source ($q = 0$), the heterogeneous problem (2.20) turns into a homogeneous problem and can be solved as an eigenvalue problem

$$\mathrm{A}\psi = \frac{1}{k_{\mathrm{eff}}}\,\mathrm{F}\psi \ , \tag{2.21}$$

with the multiplication factor k_{eff} as the eigenvalue and the neutron flux ψ as the eigenvector. This problem features a number of properties (see, e.g. Hébert, 2009), one of them being the fact that each nontrivial eigensolution can be arbitrarily normalized. The value of the normalization constant is generally computed from the thermal power P of the reactor using

$$\int_0^\infty \int_V h(\boldsymbol{r}, E)\,\phi(\boldsymbol{r}, E)\ \mathrm{d}^3 r\ \mathrm{d}E = P \ . \tag{2.22}$$

Here V is the volume of the reactor and $h(\boldsymbol{r}, E)$ is the power factor giving the recoverable energy in terms of flux.

The complicated three-dimensional geometric structure of a nuclear reactor core made of various materials that interact with neutrons in many different ways, makes the neutron transport a very difficult problem to solve. In a typical solution scheme, the problem is broken down into smaller subproblems that are solved in a stepwise fashion (see the next section), while employing many simplifications and approximations[20] (e.g., discretization in space, energy, angle, and time), justified by the concept of equivalency.

One of the important approximations usually used in the transport theory for angular treatment of the scattering cross section $\Sigma_{\mathrm{s}}(\boldsymbol{r}, E \leftarrow E', \boldsymbol{\Omega} \leftarrow \boldsymbol{\Omega}')$ in isotropic media is its expansion in terms of real spherical harmonics truncated at order L[21]. If applied, it is possible to rewrite the first term on the right side of Eq. (2.17) as

$$\int_0^\infty \sum_{l=0}^{L} \frac{2l+1}{4\pi} \Sigma_{\mathrm{s}\,l}(\boldsymbol{r}, E \leftarrow E') \sum_{m=-l}^{l} R_l^m(\boldsymbol{\Omega})\,\phi_l^m(\boldsymbol{r}, E')\ \mathrm{d}E' \ , \tag{2.23}$$

where $\Sigma_{\mathrm{s}\,l}(\boldsymbol{r}, E \leftarrow E')$ are the Legendre coefficients defined using the Legendre polynomials $P_l(\mu)$, functions of direction cosines μ:

$$\Sigma_{\mathrm{s}\,l}(\boldsymbol{r}, E \leftarrow E') = \int_{-1}^{1} \Sigma_{\mathrm{s}}(\boldsymbol{r}, E \leftarrow E', \mu)\,P_l(\mu)\ \mathrm{d}\mu \ , \tag{2.24}$$

[20] A systematically organized list of all main approximations was prepared by Reuss (2008).

[21] $L = 0$ corresponds to isotropic scattering and $L = 1$ to linearly anisotropic scattering in the laboratory frame of reference. $L = 1$ or 3 are usually satisfactory for core calculations, whereas $L \geq 5$ are required for propagation or radiation shielding calculations.

and $\phi_l^m(\boldsymbol{r}, E')$ are the spherical harmonics components of the flux defined using the real spherical harmonics components $R_l^m(\boldsymbol{\Omega})$:

$$\phi_l^m(\boldsymbol{r}, E) = \int_{4\pi} R_l^m(\boldsymbol{\Omega})\, \psi(\boldsymbol{r}, E, \boldsymbol{\Omega})\, \mathrm{d}^2\Omega \ . \tag{2.25}$$

Assuming the multi-group discretization in energy, that is, dividing the energy domain into a set of N_g (mono)energy groups characterized by the group-averaged values of the neutron flux and cross sections that preserve the reaction rates[22], the differential steady-state transport equation (2.16) transforms into the multi-group form

$$\boldsymbol{\Omega} \cdot \nabla \phi_g(\boldsymbol{r}, \boldsymbol{\Omega}) + \Sigma_{\mathrm{t}\,g}(\boldsymbol{r})\, \phi_g(\boldsymbol{r}, \boldsymbol{\Omega}) = Q_g(\boldsymbol{r}, \boldsymbol{\Omega}), \quad g = 1, 2, \ldots, N_\mathrm{g} \tag{2.26}$$

and the multi-group source density becomes

$$\begin{aligned} Q_g(\boldsymbol{r}, \boldsymbol{\Omega}) = &\sum_{h=1}^{N_\mathrm{g}} \sum_{l=0}^{L} \frac{2l+1}{4\pi} \Sigma_{\mathrm{s}\,l\,g\leftarrow h}(\boldsymbol{r}) \sum_{m=-l}^{l} R_l^m(\boldsymbol{\Omega})\, \phi_{l\,h}^m(\boldsymbol{r}) \\ &+ \frac{1}{4\pi\, k_\mathrm{eff}} \sum_{i=1}^{N_\mathrm{fiss}} \chi_{i\,g} \sum_{h=1}^{N_\mathrm{g}} \nu_{i\,h}\, \Sigma_{\mathrm{f}\,i\,h}(\boldsymbol{r})\, \phi_h(\boldsymbol{r}) \ , \end{aligned} \tag{2.27}$$

for each energy group g. In Eq. (2.27),

$\phi_g(\boldsymbol{r})$	is the neutron flux in group g,
$\Sigma_{\mathrm{t}\,g}(\boldsymbol{r})$	is the macroscopic total cross section in group g,
$\Sigma_{\mathrm{s}\,l\,g\leftarrow h}(\boldsymbol{r})$	is the Legendre coefficient for group g,
$\phi_{l\,h}^m(\boldsymbol{r})$	is the spherical harmonics components of the flux in group h,
$\chi_{i\,g}(\boldsymbol{r})$	is the fission spectrum in group g for fissile nuclide i,
$\nu_{i\,h}(\boldsymbol{r})$	is the average number of neutrons emitted per one fission of nuclide i in group h, and
$\Sigma_{\mathrm{f}\,i\,h}(\boldsymbol{r})$	is the macroscopic fission cross section for nuclide i in group h.

In practice, however, the transport equation is frequently solved by assuming the isotropy of the scattering sources in the laboratory system. This approximation is generally not valid but can be mitigated by performing a transport correction on the cross sections appearing in the transport equation, that is, by adding a forward-peaked component in the Legendre expansion of the differential scattering cross section (Hébert, 2009).

The solution of the neutron transport equation is the central problem of the part of reactor physics commonly called neutronics. Many techniques have

[22] Group condensation.

been developed for solving the neutron transport equation. Among the most widely used ones are the method of spherical harmonics (P_N), the collision probability method (CPO), the discrete ordinates method (S_N), the method of characteristics (MOC), and the Monte Carlo method (e.g., Hébert, 2009). Each of these techniques is usually suited only for certain applications like resonance self-shielding calculation, or assembly calculation. In case of large reactor core calculations, standard numerical analysis techniques such as finite-difference, finite-element, or nodal methods are normally used to solve the steady-state diffusion equation, an approximate form of the transport equation.

Diffusion equation. The diffusion equation is obtained by substituting Fick's law in the angle-integrated transport equation. Fick's law is a heuristic formula that relates the integrated neutron flux $\phi(\boldsymbol{r}, E)$ with the net integrated neutron current $\boldsymbol{J}(\boldsymbol{r}, E)$ via the diffusion coefficient $D(\boldsymbol{r}, E)$:

$$\boldsymbol{J}(\boldsymbol{r}, E) = -D(\boldsymbol{r}, E)\, \nabla\phi(\boldsymbol{r}, E) \ . \tag{2.28}$$

Similarly to the integrated neutron flux, the integrated current $\boldsymbol{J}(\boldsymbol{r}, E)$ is defined as

$$\boldsymbol{J}(\boldsymbol{r}, E) = \int_{4\pi} \boldsymbol{J}(\boldsymbol{r}, E, \boldsymbol{\Omega})\ \mathrm{d}^2\Omega \ , \tag{2.29}$$

where $\boldsymbol{J}(\boldsymbol{r}, E, \boldsymbol{\Omega})$ is the neutron angular current:

$$\boldsymbol{J}(\boldsymbol{r}, E, \boldsymbol{\Omega}) = \boldsymbol{\Omega}\, \psi(\boldsymbol{r}, E, \boldsymbol{\Omega}) \ . \tag{2.30}$$

Following the diffusion theory, the value of the proportionality coefficient $D(\boldsymbol{r}, E)$ can be calculated for negligible absorption ($\Sigma_\mathrm{a} \approx 0\,\mathrm{cm}^{-1}$) according to

$$D(\boldsymbol{r}, E) = \frac{1}{3\Sigma_\mathrm{t}(\boldsymbol{r}, E)} \ . \tag{2.31}$$

Utilizing the multi-group discretization, the multi-group steady-state diffusion equation for a critical system can be written as:

$$\begin{aligned} -\nabla\cdot\mathbb{D}_g(\boldsymbol{r})\,\nabla\phi_g(\boldsymbol{r}) + \Sigma_{\mathrm{t}\,g}(\boldsymbol{r})\,\phi_g(\boldsymbol{r}) = \\ \sum_{h=1}^{N_\mathrm{g}} \Sigma_{g\leftarrow h}(\boldsymbol{r})\,\phi_h(\boldsymbol{r}) + \frac{1}{k_\mathrm{eff}} \sum_{i=1}^{N_\mathrm{fiss}} \chi_{i\,g} \sum_{h=1}^{N_\mathrm{g}} \nu_{i\,h}\,\Sigma_{\mathrm{f}\,i\,h}(\boldsymbol{r})\,\phi_h(\boldsymbol{r}), \\ g = 1, 2, \ldots, N_\mathrm{g} \ , \end{aligned} \tag{2.32}$$

where $\mathbb{D}_g(\boldsymbol{r})$ is a 3×3 diagonal tensor containing directional diffusion coefficients in energy group g, $\Sigma_{g\leftarrow h}(\boldsymbol{r})$ is the macroscopic scattering cross section from group h to group g, and the remaining notation is standard.

One can write for the normalization of the eigensolution:

$$\sum_{g=1}^{N_\mathrm{g}} \int_V h_g(\boldsymbol{r})\, \phi_g(\boldsymbol{r})\, \mathrm{d}^3 r = P \ . \tag{2.33}$$

Solution of the diffusion equation for a heterogeneous reactor composed of several different regions is done using neutron flux and current continuity conditions at the interface between each two adjacent regions. Let $\mathcal{I}$ be an interface between two neighboring regions, $\boldsymbol{N}(\boldsymbol{r})$ a unit vector, located in a point $\boldsymbol{r}$, normal to $\mathcal{I}$, $\epsilon \in \mathbf{R}^+$ an infinitesimally small positive real number, and $\boldsymbol{r}^\pm \equiv \boldsymbol{r} \pm \epsilon\, \boldsymbol{N}(\boldsymbol{r})$. Then the flux continuity condition can be expressed as

$$\phi_g(\boldsymbol{r}^+) = \phi_g(\boldsymbol{r}^-), \quad \boldsymbol{r} \in \mathcal{I} \ , \tag{2.34}$$

and the current continuity condition as

$$\boldsymbol{J}_g(\boldsymbol{r}^+) \cdot \boldsymbol{N}(\boldsymbol{r}) = \boldsymbol{J}_g(\boldsymbol{r}^-) \cdot \boldsymbol{N}(\boldsymbol{r}), \quad \boldsymbol{r} \in \mathcal{I} \ . \tag{2.35}$$

Neutron leakage is accurately accounted for by setting up conditions for neutrons effectively escaping the reactor spatial domain V through its real boundary ∂V. Two basic types of boundary conditions exist in diffusion theory; the zero-flux boundary condition

$$\phi_g(\boldsymbol{r}) = 0, \quad \boldsymbol{r} \in \partial V \ , \tag{2.36}$$

and the more exact albedo boundary condition

$$\mathbb{D}_g(\boldsymbol{r})\, \nabla\phi_g(\boldsymbol{r}) \cdot \boldsymbol{N}(\boldsymbol{r}) + \frac{1}{2}\frac{1-\beta_g(\boldsymbol{r})}{1+\beta_g(\boldsymbol{r})}\, \phi_g(\boldsymbol{r}) = 0, \quad \boldsymbol{r} \in \partial V \ , \tag{2.37}$$

where $\boldsymbol{N}(\boldsymbol{r})$ is a unit vector, located at $\boldsymbol{r}$, normal to ∂V, and $\beta_g(\boldsymbol{r})$ is the albedo at $\boldsymbol{r}$ for group g, defined as the ratio between the incoming and outgoing net currents $J_g^-(\boldsymbol{r})$ and $J_g^+(\boldsymbol{r})$:

$$\beta_g(\boldsymbol{r}) = \frac{J_g^-(\boldsymbol{r})}{J_g^+(\boldsymbol{r})} \ . \tag{2.38}$$

The most usual values are $\beta_g(\boldsymbol{r}) = 0$ for all energy groups to represent a zero incoming condition and $\beta_g(\boldsymbol{r}) = 1$ to represent a symmetry condition.

The validity of the diffusion approximation is given by the validity of Fick's law (2.28) (e.g., Lamarsh, 1972; Reuss, 2008), which was derived assuming

little geometric heterogeneity and small absorption of the medium, and a position not too close to the interfaces/boundaries and concentrated sources (at least a few mean free paths far from them). Much of the error due to the diffusion approximation comes from the assumption of isotropic scattering in the laboratory system. However, most of this error can be compensated by introducing a modified version of the diffusion coefficient (2.31), obtained through the transport theory (the transport correction). In the multi-group steady-state diffusion equation, the corrected non-directional scalar diffusion coefficient in energy group g becomes

$$D_g(\boldsymbol{r}) = \frac{1}{3\left(\Sigma_{\mathrm{t}\,g}(\boldsymbol{r}) - \bar{\mu}(\boldsymbol{r})\,\Sigma_{\mathrm{s}\,g}(\boldsymbol{r})\right)} = \frac{1}{3\,\Sigma_{\mathrm{tr}\,g}(\boldsymbol{r})}\ , \tag{2.39}$$

where $\bar{\mu}(\boldsymbol{r})$ is the average cosine of the neutron deviation angle in a collision and $\Sigma_{\mathrm{s}\,g}(\boldsymbol{r})$ is the macroscopic scattering cross section in group g. The term $\Sigma_{\mathrm{tr}\,g}(\boldsymbol{r}) = \Sigma_{\mathrm{t}\,g}(\boldsymbol{r}) - \bar{\mu}(\boldsymbol{r})\,\Sigma_{\mathrm{s}\,g}(\boldsymbol{r})$ in Eq. (2.39) denotes the transport cross section.

Despite its limitations, coarse-mesh few-group diffusion codes yet remain the most common tools used within nuclear fuel management for preliminary three-dimensional whole-core neutronics analysis for both thermal and fast reactors. It should be mentioned though that more detailed neutronics evaluation of systems with a harder neutron spectrum that goes beyond the scope of preliminary analysis, must account for angular distributions of the neutron flux at each space point in the system. The need for a transport calculation may, therefore, become a necessity in case of ADSs, for instance (Aizawa et al., 2013).

Fuel Depletion

The composition of materials in a reactor core evolves during the operation as a result of particle induced transmutation reactions and radioactive decays of unstable isotopes, an example of which is illustrated in Figure 3.5. This process is important especially for the heavy nuclides present in the fuel, causing changes in their concentrations $N(t)$, and thus, also in the macroscopic cross sections $\Sigma_{\mathrm{x}}(E,t) = N(t)\,\sigma_{\mathrm{x}}(E)$, which further influence the core reactivity and the power and neutron flux distributions. Therefore, to account for these effects, it is sometimes necessary to evaluate the reactor core characteristics at different times.

The mathematical problem to be solved is described by the following system of nonlinear depletion equations[23]:

$$\frac{\partial}{\partial t} N_i(\boldsymbol{r},t) = -\sum_{\substack{j=1\\ j\neq i}}^{N_{\text{nucl}}} N_i(\boldsymbol{r},t)\left(\lambda_{j,i} + \int_0^\infty \sigma_{j,i}(\boldsymbol{r},E)\,\phi(\boldsymbol{r},E,t)\;\mathrm{d}E\right) + \sum_{\substack{j=1\\ j\neq i}}^{N_{\text{nucl}}} N_j(\boldsymbol{r},t)\left(\lambda_{i,j} + \int_0^\infty \sigma_{i,j}(\boldsymbol{r},E)\,\phi(\boldsymbol{r},E,t)\;\mathrm{d}E\right),$$

$$i = 1, 2, \ldots, N_{\text{nucl}}\ , \tag{2.40}$$

where

N_{nucl}	is the number of considered nuclides,
$N_i(\boldsymbol{r},t)$	is the concentration of a nuclide i,
$\lambda_{i,j}$	is the decay constant for the decay from a nuclide j to a nuclide i,
$\sigma_{i,j}(\boldsymbol{r},E)$	is the particle induced microscopic cross section for transmutation from a nuclide j to a nuclide i, and
$\phi(\boldsymbol{r},E,t)$	is the time-dependent neutron flux.

The length of the burnup step should be chosen so that the neutron spectrum, and hence the microscopic cross sections, are subject to little variation, and can be considered constant over the interval. If the same but more restrictive assumption is made for the flux, $\phi(\boldsymbol{r},E,t) \to \phi(\boldsymbol{r},E)$, one receives a system of linear equations.

Many techniques are available to solve the depletion equations. A brief review of them can be found, for instance, in Isotalo and Aarnio, 2011; Krüger, 2004.

2.3.2 Computational Scheme

As indicated earlier, the calculation of the neutron distribution in a reactor core is done in multiple consecutive steps that create what is known as a computational (or calculational) scheme. The practical aspects of performing neutronics calculations (mainly for LWRs) are well described by Hébert in his book Applied Reactor Physics (2009). A well-known book that has a significant part on numerical methods is Duderstadt and Hamilton, 1976. More comprehensive information on reactor analysis practices is in Ronen, 1986a,b,c.

[23] Also known as fuel evolution or fuel burnup equations. Another frequently but, in a sense, somewhat incorrectly used name is the Bateman equations (Bateman, 1910).

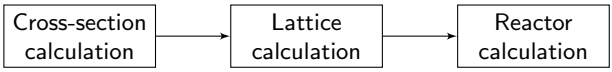


Figure 2.2: Global computational scheme for reactor core analysis

The second volume of the triplet includes also a part on fast reactor calculations by Salvatores (1986). Some general information can be also found in Reuss, 2008, and other books. In this section are described the basic principles of reactor analysis as they are an integral part of in-core nuclear fuel management.

Neutron distribution in the reactor core can be predicted using the three main steps depicted in Figure 2.2. The basic principle is to start the calculation with a very fine representation in neutron energy (with a coarse representation in space) and to terminate with a very fine representation in space (with a coarse representation in energy)(Hébert, 2009). First, a multi-group library of average neutron cross section values is prepared that reflects various neutron-nuclide interactions, resonant character of the cross sections as functions of neutron energy, and dependence on the nuclide temperature. These cross sections are then weighted with the neutron flux characteristic for a unit cell or assembly in the modeled system, providing data that can be used in full-core calculations, the last step in the global scheme.

Deterministic computer codes and analytical procedures developed to solve neutron transport related problems at either level, fuel-lattice or full-core, use various simplifying built-in assumptions and approximations, which limit their validity to a certain class of nuclear systems (e.g., BWRs versus PWRs, thermal systems versus fast systems, critical systems versus subcritical systems). Selection of a right code and its proper use therefore requires basic knowledge of these assumptions, usually originating in the characteristic physics properties of the considered nuclear system.

2.4 MYRRHA

Since its foundation in 1952, the Belgian Nuclear Research Centre (SCK•CEN) at Mol has always been involved in conception, design, realization, and operation of large nuclear infrastructures. In 1962, the BR2 high-flux light-water-cooled and moderated material testing reactor (SCK•CEN, 2011) was put into operation, an excellent research tool, which has produced remarkable results for the international nuclear energy community in various fields such as material research for fission and fusion reactors, fuel research, reactor safety, reactor

technology, and for the production of radioisotopes for medical and industrial applications. The BR2 reactor is now licensed for operating until 2016 with a potential extension for another ten-year period until 2026. Therefore, work on a new multipurpose flexible irradiation facility that could replace BR2 began in late 1990s. This facility is called MYRRHA—The Multi-purpose hYbrid Research Reactor for High-tech Applications (Aït Abderrahim and Baeten, 2012).

2.4.1 Applications

Motivated by the future needs of the nuclear sector (Section 1.1) and particular interests of the laboratory, a variety of MYRRHA applications has been envisaged in the early phase of the project with the goal (Aït Abderrahim, Kupschus, et al., 2001; Aït Abderrahim and Baeten, 2012):

- to demonstrate the ADS concept by coupling an accelerator, a spallation target, and a subcritical reactor at a reasonable power level to allow an operation feedback, scalable to an industrial demonstrator,
- to allow study of the efficient transmutation of HLW, in particular minor actinides, that would request high fast flux intensity,
- to be operated as a flexible fast spectrum irradiation facility allowing for fuel developments for innovative reactor systems, for material developments for Generation IV systems (Section 1.1.2) and for fusion reactors, and finally, for radioisotope production for medical and industrial applications.

In order to make it possible to fulfill its mission, meet the objectives, and achieve the required irradiation performance levels (see Table 2.1), the MYRRHA project has been evolving towards the current reactor design through several stages briefly described in the following section.

[24]The threshold of $E = 0.75\,\text{MeV}$ is a representative value on average for the cross-over point of $\sigma_\text{f}(E)$ and $\sigma_\text{c}(E)$ for most important MAs (e.g., $E = 0.48\,\text{MeV}$ for ^{237}Np, $E = 0.73\,\text{MeV}$ for ^{241}Am, $E = 0.39\,\text{MeV}$ for ^{244}Cm). This threshold therefore indicates when fission becomes the dominant reaction for the considered nuclides (Aït Abderrahim, Al Mazouzi, et al., 2012). *Note:* The values were calculated based on the ENDF/B-VII.1 data library.

[25]The threshold of $E = 1\,\text{MeV}$ is used by most material specialist to indicate the fast flux responsible for material damage; alternatively the threshold of $E = 0.1\,\text{MeV}$ is sometimes used by other groups of material specialists (Aït Abderrahim, Al Mazouzi, et al., 2012).

Table 2.1: MYRRHA applications and required irradiation levels

Application	Irradiation level	
	E range	ϕ ($\mathrm{n\,cm^{-2}\,s^{-1}}$)
HLW transmutation research	$> 0.75\,\mathrm{MeV}$[24]	10^{15}
Generation IV fuel development	total	5×10^{14} to 10^{15}
Generation IV material development	$> 1\,\mathrm{MeV}$[25]	1 to 5×10^{14}
Fusion reactor material development	$> 1\,\mathrm{MeV}$	1 to 5×10^{14}
Radioisotope production	thermal	2 to 3×10^{15}
Silicon doping	thermal	10^{13} to 10^{14}

Source: Aït Abderrahim, Al Mazouzi, et al., 2012.

2.4.2 Project Evolution

MYRRHA started as a follow-up of ADONIS (1995–1997), the first project at SCK•CEN, where a small-scale ADS was studied in collaboration with IBA (Ion Beam Applications, Louvain-la-Neuve, Belgium), with a single objective to produce radioisotopes for medical purposes, ^{99m}Tc/^{99}Mo in particular (Jongen, 1999; Stichelbaut and Jongen, 2011). In 1998, it was decided to extend the purpose of ADONIS to become a material testing reactor for material and fuel research, to study the feasibility of transmutation of minor actinides, and to demonstrate the principle of the ADS at a reasonable power level. Since then, the project is named MYRRHA (Baeten et al., 2011). Currently, the project is in the pre-licensing phase, which should be followed by the construction (2017–2021) and commissioning (2022–2024) periods. It is foreseen that the facility will be operational at full power around 2025 (SCK•CEN, 2014).

The first conceptual design of MYRRHA was developed during the years 1998–2005 within the PDS-XADS initiative (Aït Abderrahim, Kupschus, et al., 2001; Aït Abderrahim, De Bruyn, and al., 2007; Giraud et al., 2005; Maes, 2005). The project then continued between the years 2005 and 2010 and resulted in an updated design called XT-ADS (Aït Abderrahim, De Bruyn, and Giot, 2007; De Bruyn, Larmignat, et al., 2010; De Bruyn and Maes, 2007). XT-ADS served as a starting point for the latest continuation of the project, which has evolved into the last official version known as MYRRHA-FASTEF (Aït Abderrahim, Al Mazouzi, et al., 2012; Aït Abderrahim and al., 2012; Aït Abderrahim and Baeten, 2012; De Bruyn, Baeten, et al., 2010; De Bruyn, Aït Abderrahim, et al., 2014; De Bruyn, Engelen, et al., 2012; De Bruyn, Fernandez, et al., 2012; Sarotto, Castelliti, et al., 2013). Among the biggest changes in the MYRRHA-FASTEF design comparing to XT-ADS, which led to an improved neutronics

performance and an overall simplification of the reactor core, and thereby, to the minimization of technological risks, are an increased number of fuel assemblies, a higher proton energy, and the use of a loop-less T91 window spallation target, now hosted only in one assembly position in the center of the core instead of three previously occupied positions. Unlike XT-ADS, the new design also allows to operate the reactor as a critical system. It is the MYRRHA-FASTEF version, which will be used as a reference design in this dissertation.

In the two following sections, the main MYRRHA-FASTEF technology choices will be justified and principal parts of the facility will be shortly described, with a focus on the reactor core. A description of other important parts like the primary, secondary, and tertiary cooling systems, buildings, or a detailed description of the accelerator technology will be omitted here as they do not relate to the subject of this work. A comprehensive information on the whole facility is included in the report by Aït Abderrahim, Al Mazouzi, et al. (2012). A detailed description of the MYRRHA-FASTEF critical and subcritical cores can be found in the article by Sarotto, Castelliti, et al. (2013) and also in the report by Sarotto, Fernandez, et al. (2012). Main design parameters are summarized in Table 2.2. The data on the irradiation performance of the machine are summarized in Table 2.3 and Table 2.4.

2.4.3 Technology Choices

Accelerator and Spallation Target

A particle accelerator and a spallation target are essential parts of an ADS (see Section 1.1.2). The MYRRHA accelerator provides protons accelerated up to 600 MeV, which hit the surface of the liquid lead-bismuth eutectic (LBE) coolant (spallation target), delivering extra high-energy neutrons into the system that are needed to sustain the chain reaction in the subcritical core. The maximum proton beam intensity of 4 mA was determined based on the neutron flux requirements and the subcritical core design.

One of the challenging engineering aspects of the ADS technology has been related to the stringent requirements on the accelerator reliability and availability expressed in terms of the mean time between failures (MTBF). In the MYRRHA case, the beam trip duration tolerance is three seconds, because longer beam trips may initiate thermal shocks with a very negative effect on the system. Within the operational period of MYRRHA, the number of allowed beam trips exceeding three seconds must remain under 10, shorter beam trips are allowed without limitations (Aït Abderrahim, De Bruyn, Van den Eynde, et al., 2014).

Table 2.2: MYRRHA-FASTEF design parameters

Accelerator	
Type	proton linac
Maximum current	4 mA
Beam energy	600 MeV
Minimum MTBF	250 h
Fuel pin	
Fuel type	MOX
Fuel enrichment[a]	30–35 wt%
Cladding material	15-15Ti
Fuel active height	60 cm
Fuel assembly	
P/D	1.2–1.3
Number of pins	127
Wrapper material	T91
Wire spacer material	15-15Ti
Maximum LBE bulk speed	$2\,\mathrm{m\,s^{-1}}$
Core	
Maximum power	$100\,\mathrm{MW_{th}}$
Number of positions	151
Number of MFCs	37
Core diameter	≈ 1.5 m
Maximum core pressure drop	2.5 bar
Reactor vessel	
Inner diameter	≈ 8 m
Length	≈ 12 m
Material	AISI 315L
Temperatures	
Cold shutdown state	200 °C
Maximum core inlet temperature	270 °C
Average core outlet temperature	410 °C
Maximum fuel cladding temperature	550/466 °C
Maximum admissible reactor power	$110\,\mathrm{MW_{th}}$
LBE inventory	≈ 4320 t

[a] Mixture of PuO_2 from spent fuel and natural UO_2.

Note: Values of some parameters are not precisely specified for confidentiality reasons.

Source: Aït Abderrahim, Al Mazouzi, et al., 2012.

Table 2.3: MYRRHA-FASTEF irradiation performance

Application	Criterion	Performance
^{99}Mo production	radioactivity	$\approx 250\,\mathrm{Ci\,g_U^{-1}}$
Silicon doping	thermal ϕ	4 to $15 \times 10^{12}\,\mathrm{n\,cm^{-2}\,s^{-1}}$
MA transmutation[a]	transm. rate	$\approx 32\,\mathrm{kg_{HM}\,TW^{-1}\,h^{-1}}$
Material irradiation	DPA damage	$\approx 30\,\mathrm{dpa\,y^{-1}}$

[a] The values were obtained by inserting EFIT-like FAs (Artioli et al., 2008) into the MYRRHA IPS positions.

Source: Aït Abderrahim, Al Mazouzi, et al., 2012.

A linear accelerator (linac) that operates in a continuous wave mode was selected for MYRRHA as a preferred option over a sector-focused cyclotron, the only alternative solution with megawatt level beam power. A linac has the potential for implementing a fault tolerance scheme and offers a high modularity, resulting in the possibility to recover the beam within a short time and increasing the beam energy. The combination of redundancy and fault tolerance should allow obtaining the MTBF value in excess of required 250 h (Aït Abderrahim, De Bruyn, Van den Eynde, et al., 2014). The major drawback of a linac is its length, which depends on the final beam energy and accelerating gradient. This length is an important factor in the cost of the facility, since most of the accelerator needs to be shielded against radiation.

Subcriticality level. The sub-criticality level of around $k_{\mathrm{eff}} = 0.95$ has been considered as an appropriate level for MYRRHA-FASTEF. This is the criticality level accepted by the safety authorities for fuel storage and would allow the system to remain subcritical even when accounting for the possible positive reactivity injections (Aït Abderrahim, Al Mazouzi, et al., 2012).

Fuel

MOX fuel has been chosen for MYRRHA due to the large experience with this type of fuel in Europe and in Belgium in particular. A maximum plutonium enrichment of 35 wt% was first considered based on the available manufacturing and qualification experience by Belgonucleaire, a former Belgian MOX fuel producer and the founder of the technology (Aït Abderrahim, Al Mazouzi, et al., 2012). However, in the later phase of the project, it was decided to decrease the plutonium mass in the fuel content down to 30 wt% to make it more feasible from the manufacturing point of view, and also, to obtain more relaxed power-peaking

factors for the subcritical core. The complement of the Pu content in the fuel pellet consists of natural uranium. Pu and Am compositions are derived from the reprocessing of PWR spent fuel of 45 GW d t_{HM}^{-1} burnup (initially enriched with 4.5 wt% of ^{235}U) and a cooling period of 50 years. The spontaneous decay of fissile nuclides, as ^{241}Pu, is taken into account by introducing the 1.65 wt% of Am into the Pu vector (Sarotto, Castelliti, et al., 2013).

Coolant

Three materials were identified as prospective coolants for Generation IV fast spectrum systems (Section 1.1.2): helium (GFR), sodium (SFR), and lead (LFR). Gaseous coolants have been excluded from the consideration for MYRRHA because of the desired high fast flux, the high power density, and the short term deployment of the facility. The use of sodium has also been found undesirable due to safety related issues, despite the fact that most of the operation experience with the fast reactor technology has been made with SFRs. The vigorous chemical reaction of sodium and oxygen from water or air imposes a high risk of fire, which is unacceptable especially in case of a flexible irradiation facility like MYRRHA, where a lot of loading/unloading operations of experiments occur during the reactor operation. Out of the two remaining liquid metal alternatives, which are lead and LBE, the latter was selected for its low melting temperature 124.5 °C, which permits to decrease the core inlet temperature to 270 °C. This allows to reduce the risk of corrosion and to increase the average temperature gradient over the fissile zone up to $\Delta T = 140$ °C in nominal conditions (Aït Abderrahim, Al Mazouzi, et al., 2012).

Structural Materials

Most of the primary system structures in MYRRHA are made of AISI 316L stainless steel. The fuel assembly hexagonal wrapper, the core support plate, the in-pile section guide tube, and the shutdown systems are made of T91 ferritic-martensitic steel, because of its higher allowable stresses, less swelling under irradiation, and better creep resistance than AISI 316L. Austenitic 15-15Ti stainless steel (15Ni-15Cr stainless steel stabilized by Ti) was chosen as a material for the fuel cladding for its high resistance to embrittlement. This steel is also well known and has already been qualified as a fuel cladding for SFRs, making the MYRRHA fuel licensing process easier by reducing the required short term research load (Sarotto, Castelliti, et al., 2013).

2.4.4 Reactor Design

As can be seen from Figure 2.3. the MYRRHA reactor is a complex structure consisting of many components. The subject of this work, however, concentrates only on a small part of the system, which is the reactor core composed of fuel, reactor control and shutdown elements, experimental rigs, and spallation target. These parts will be described in a greater detail in Chapter 3.

Reactor

In order to profit from the thermal inertia provided by a large coolant volume ($\approx 4320\,\mathrm{t}$), the MYRRHA-FASTEF reactor was designed as a pool-type reactor with all primary systems housed inside a large reactor vessel. The vessel is covered by a thick structure that provides a basis for all inner-vessel components, including two primary pumps, four primary heat exchangers, two fuel handling machines, two silicon doping facilities, experimental rigs, the beam line, and the core barrel, which holds the core support plate, the core jacket, and the reactor core itself. The control rods, the safety rods, the ^{99}Mo productions units, and the beam line are inserted into the reactor core from above through the core plug. Another important part of the design is the diaphragm, which provides the physical barrier between the hot and the cold coolant and also the supporting structure for special compartments that serve as the inner-vessel fuel storage.

Fuel loading. The fuel loading is done from underneath to keep a large flexibility for the experimental devices inserted from the top of the core. Analogously to conventional reactors, where fuel loading is assisted by gravity, fuel loading in MYRRHA is assisted by buoyancy force.

Power level. In order to attain the demanded irradiation parameters, the maximum MYRRHA-FASTEF core power was set to 100 MW. The primary, secondary, and tertiary cooling systems were designed to evacuate 110 MW. The additional 10 MW capacity accounts for the power coming from various sources like the decay heat of the spent fuel in the in-vessel storage, the primary pumps, the decay heat of the activation[26] and spallation products in the coolant, the γ-heating in structures and the contribution of the proton beam in the

[26] An important activation product in lead-bismuth-cooled FRs is ^{210}Po, a strong emitter of high-energy α-particles (5.3 MeV), produced by neutron activation of ^{209}Bi. Because polonium forms a very volatile hydride upon contact with water, its extraction from the Pb-Bi coolant is essential to minimize the risk of its transport out of the reactor pool.

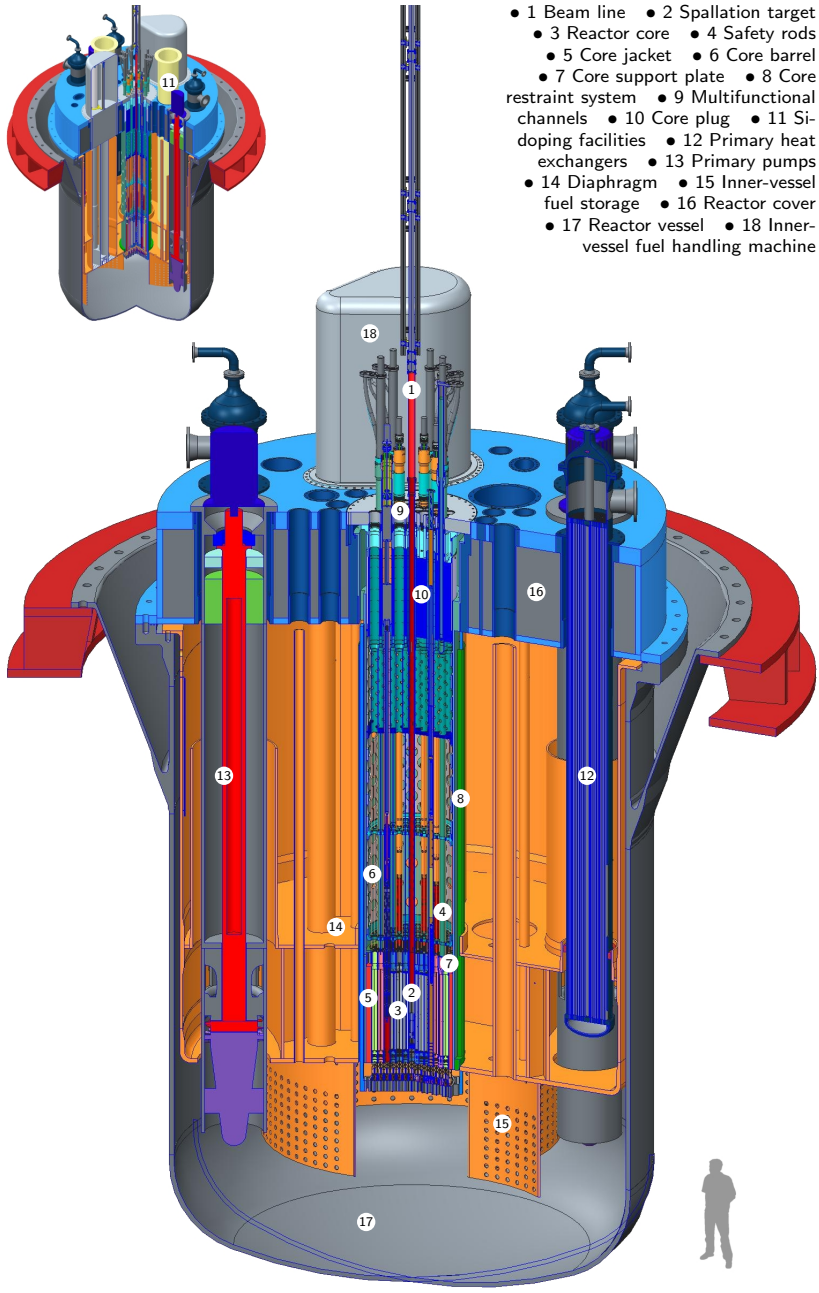


Figure 2.3: MYRRHA-FASTEF reactor

subcritical mode (Sarotto, Castelliti, et al., 2013). In a real operation, however, the reactor power will have to be reduced to 72 MW and 60 MW for the critical and the ADS mode, respectively, due to the current limitations in the LBE conditioning system.

Working temperatures. Unlike in other reactors, where the maximum operating temperatures are constrained by the thermal-mechanical properties of the fuel, in MYRRHA, the maximum range of the working temperatures is determined by the physical properties of the fuel cladding material (15-15Ti SS) and the precision of the LBE conditioning system in controlling the dissolved oxygen content in the coolant.

In SFRs, an accurate oxygen control is used to mitigate the corrosive effects of the coolant on the fuel elements. For a given temperature, the concentration of the dissolved oxygen should be high enough to form oxide layers on the steel surfaces that protect them from further dissolution. It also makes the oxide layers recover in case of erosion by the flowing LBE, At the same time, the oxygen concentration should be low enough to prevent the formation of lead oxide, which can plug some reactor components such as the heat exchangers or the fuel assemblies (Aït Abderrahim, Al Mazouzi, et al., 2012; Lim, 2011).

Given the decided cold shutdown temperature 200 °C well above the freezing point and a conditioning margin, the maximum cladding temperature for MYRRHA should not exceed 466 °C (Aït Abderrahim, Al Mazouzi, et al., 2012). Thus, the maximum cladding temperature constitutes the most stringent limiting factor with respect to the operating temperatures rather than the fuel melting point[27], which is implicitly met in this case.

The maximum cladding temperature corresponding to 100 MW power is ≈ 550 °C. It is supposed that such a higher temperature could be sustained in the future, using a new coated fuel cladding that will be qualified during the facility operation. From that perspective, FASTEF may be viewed as an envelope for MYRRHA.

Reactor Core

The MYRRHA-FASTEF subcritical reactor core is composed of the fuel assemblies, dummy assemblies (wrappers filled by LBE), reflector assemblies

[27] The melting temperature of the MYRRHA fuel is approximately 2700 °C (Todreas and Kazimi, 1990a). The maximum fuel temperature 2100 °C, corresponding to the maximum linear power of $\approx 370\,\mathrm{W\,cm^{-1}}$, was used earlier as the operating temperature limit for the XT-ADS (Sarotto, Vanmaercke, et al., 2010; Van den Eynde, Nishihara, et al., 2009).

containing pins with yttria-stabilized-zirconia pellets, in-pile sections, and a central subassembly with the spallation target, all arranged in a hexagonal grid of 151 core positions. The critical core can be obtained by removing the central spallation assembly and by inserting the control rods, the safety rods, and a few more fuel assemblies into the core. The reactor core layouts used for analysis of both critical and subcritical variants are depicted in Figure 2.4. 37 of the positions are multifunctional channels (MFCs), which can be accessed from the top of the reactor and loaded by different types of assemblies, for instance, by the in-pile sections with experimental irradiation rigs. This provides the reactor the needed flexibility.

A more detailed description of the MYRRHA-FASTEF reactor core is included Chapter 3. The basic neutronics characteristics of the reference MYRRHA-FASTEF cores from Figure 2.4 are summarized in Table 2.4.

Fuel Assembly

The fuel assembly design used in MYRRHA-FASTEF is similar to a typical design used in fast reactors cooled by sodium. Each fuel assembly contains a hexagonal bundle of 127 cylindrical pins; of them, 126 are fuel pins and the central one is the stiffness pin made of steel. Each fuel pin contains fuel pellets and a gas plenum. The pins are surrounded by a hexagonal wrapper. The upper and lower end of the wrapper are connected to the inlet and outlet nozzles guiding the coolant through the fuel assembly.

In order to simplify the licensing of the MYRRHA-FASTEF fuel, several features have been adopted from the fuel assembly design of the French Super Phénix SFR, like, for instance, the type of the cladding material or the use of the helical wire-spacers to keep the pins separated one from another (Aït Abderrahim, Al Mazouzi, et al., 2012; Sarotto, Castelliti, et al., 2013).

Reactor Control and Shutdown System

The problem of the reactor control is very simplified for subcritical systems, since in an ADS, the accelerator plays the role of both control and safety systems. This is possible, because a subcritical reactor can be put into a safe state by instantaneously turning off the accelerator (Section 1.1.2), and also, because the reactor power can be controlled by modulating the beam intensity. Taking into account that absorber devices must be present in the core in order to meet the safety requirements during refueling, six control rods were positioned on

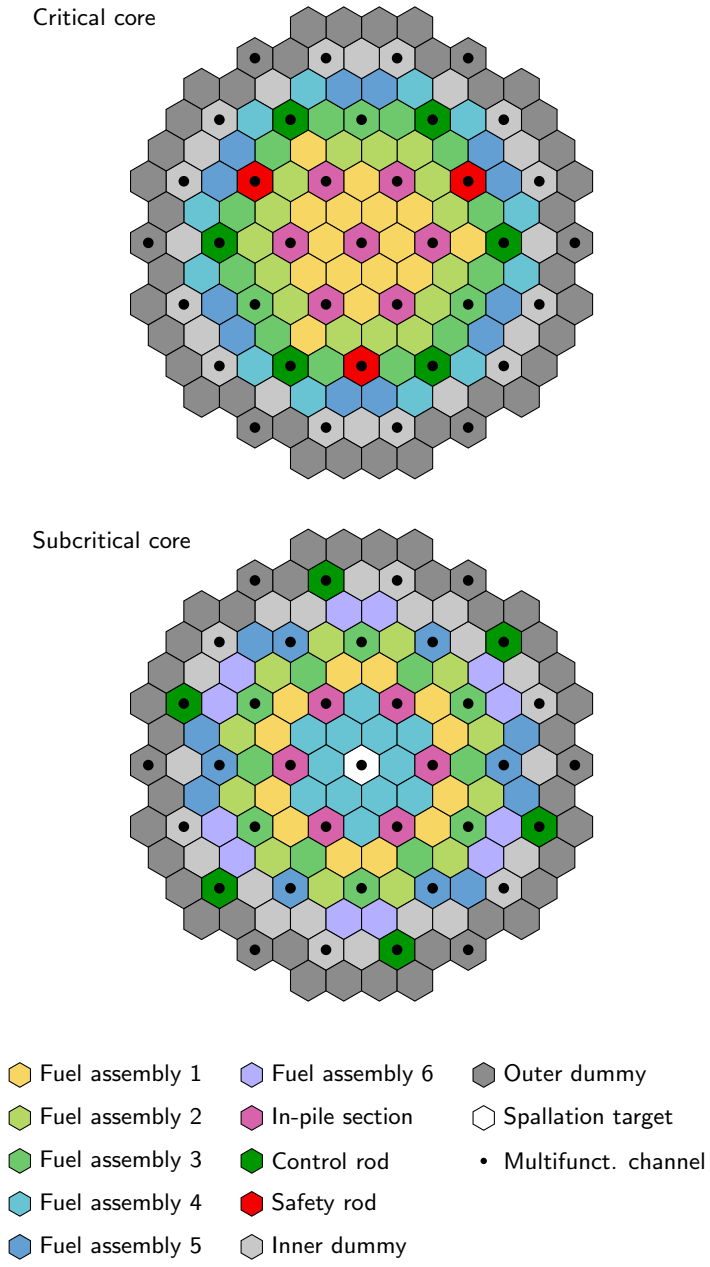


Figure 2.4: MYRRHA-FASTEF critical and subcritical reactor core layouts

Table 2.4: MYRRHA-FASTEF core characteristics

	Core			
Parameter	Critical		Subcritical	
Number of FAs	69		72	
Number of IPSs	7		6	
Number of CRs	6		6	
Number of SRs	3		0	
Enrichment (wt%)	34.5		30.0	
Power[a] (MW)	100		94	
Refueling scheme	in-out		mixed	
Fuel batches (FA)	$15+15+15+12+12$		6×12	
Characteristic[b]	BOEC	EOEC	BOEC	EOEC
k_{eff} ($\mu \pm \sigma$)	1.01580	1.00034	0.96677	0.95191
	± 0.00011	± 0.00004	± 0.00016	± 0.00015
k_{src} ($\mu \pm \sigma$)	–	–	0.97345	0.96014
			± 0.00259	± 0.00257
Beam current (mA)	–	–	2.23	3.46
q_{max} (W cm^{-1})	340.7		321.9	356.9
Power peaking				
Hot FA $P_{\text{r FA}}$	1.02		1.07	1.10
Hot FA P_{z}	1.15		1.25	1.30
$P_{\text{r core}}$	1.51		1.40	1.45
P_{q}	1.77		1.87	2.07
Peak ϕ in IPSs ($\text{n cm}^{-2}\,\text{s}^{-1}$)				
$E > 0.75\,\text{MeV}$	0.63×10^{15}		0.53×10^{15}	0.56×10^{15}
All E	3.79×10^{15}		3.33×10^{15}	3.46×10^{15}
Disch. burnup ($\text{MW d t}_{\text{HM}}^{-1}$)	–	37.3	–	44.5

[a] Corresponds to the maximum cladding temperature limit 550 °C.
[b] MCNPX/ALEPH results.
Source: Sarotto, Fernandez, et al., 2012.

the outskirts of the core, completely withdrawn during the normal operation (Sarotto, Fernandez, et al., 2012).

In case of the critical design, a new control and shutdown system had to be introduced that maintains the core critical throughout the whole cycle and, at the same time, have the anti-reactivity worth large enough to compensate the temperature decrease between the nominal conditions and the safe shutdown state, and to compensate the core geometry modifications, resulting from accidental events and handling errors during fuel manipulation ($\Delta k_{\text{eff}} \approx 2000\,\text{pcm}$).

Two fully independent systems are used in the MYRRHA-FASTEF core that ensure safe operation of the facility. The primary system has a double function (control and scram) and is implemented by buoyancy driven control rods, inserted into the core from the bottom. The control rods are placed at the core boundary to limit the flux perturbations during the operation. They are partially inserted at the beginning of cycle ($\approx 10\,\text{cm}$ in the fissile zone) and progressively extracted until the end of cycle to compensate the k_{eff} swing ($\approx 1500\,\text{pcm}$). Six control rods were foreseen to limit the reactivity insertion in case of a single rod ejection. At the beginning of a cycle, that is, in the most conservative case, this insertion must be lower than the delayed neutron fraction $\beta \approx 320\,\text{pcm}$. The control rods must also compensate for power regulation ($\approx 1000\,\text{pcm}$) and for unavoidable uncertainties as fuel fabrication errors, code and library uncertainties, etc. ($\approx 2000\,\text{pcm}$). Consequently, the total required anti-reactivity worth of the control rods must be greater than 6500 pcm (Sarotto, Fernandez, et al., 2012).

The secondary system, used only for scram, is implemented by gravity driven safety (scram) rods, fully extracted during the operation and fully inserted downward in case of an emergency shutdown. The number and positions of the safety rods were selected to withstand single failure events. The needed anti-reactivity worth $\approx 2000\,\text{pcm}$ is obtained by three safety rods located in the middle zone of the core (Sarotto, Fernandez, et al., 2012). The performance of the shutdown system is summarized in Table 2.5.

The absorber rods of both reactor shutdown systems are made of 90 % ^{10}B enriched boron carbide B_4C.

2.4.5 Nuclear Fuel Management

The nuclear fuel management of a small-scale research reactor like MYRRHA is significantly simplified compared to that of large-scale power production facilities. In this section, the most important aspects of the MYRRHA nuclear

Table 2.5: Anti-reactivity worth of the MYRRHA-FASTEF critical core shutdown system

Position		BOEC		EOEC	
CRs (6/6)	SRs (#/3)	k_{eff}	Δk_{eff} (pcm)	k_{eff}	Δk_{eff} (pcm)
Out	Out (3)	1.03075	1571	1.01615	0
10 cm in	Out (3)	1.01504	0	–	–
10 cm in	In (3)	0.96353	−5151	–	–
10 cm in	In (2)	0.98228	−3276	–	–
Out	In (3)	–	–	0.96236	−5379
Out	In (2)	–	–	0.98201	−3414
In	Out (3)	0.92712	−8792	0.91385	−10 230
In	In (3)	0.88944	−12 560	0.87665	−13 950

Source: Sarotto, Castelliti, et al., 2013 (ERANOS results).

fuel management will be commented, including the out-of-core and in-core fuel management and related optimization problems.

Out-of-Core Nuclear Fuel Management

The out-of-core fuel management analysis of MYRRHA-FASTEF was performed as a part of the reactor design process. The objective of the analysis was to achieve the desired irradiation levels in the reactor core (Table 2.1) with only one type of fuel described in Section 2.4.3 and with the limits on the maximum core power, maximum cladding temperature, and shutdown margins described in Section 2.4.4[28]. An additional constraint was imposed on the system availability by adopting the XT-ADS refueling schedule for both critical and subcritical modes: a 90-day subcycle of normal operation followed by a 30-day maintenance period and one long 90-day maintenance period after every third cycle (Sarotto, Castelliti, et al., 2013; Van den Eynde, Nishihara, et al., 2009). It was also desired to keep the number of fuel assemblies in the core as low as possible in order to reduce its size, and thus, to minimize the cost of the installation[29].

[28] Other parameters like the limit on the maximum DPA in the core barrel had also to be taken into account in the reactor design process.

[29] In the later phase of the project (after the MYRRHA-FASTEF), additional economically motivated constraints had to be considered in the out-of-core decision making process, including a maximum number of fresh fuel assemblies loaded in the reactor core at the beginning of each cycle, a minimum fuel discharge burnup, and decreased feed fuel enrichment. The new requirements led to different core designs characterized by a larger number of FAs (more than 100), an increased number of fuel batches, and a decreased number of IPSs.

Two reactor core layouts depicted in Figure 2.4 were proposed for the critical and subcritical modes that meet all the criteria. An in-out shuffle was envisaged for the critical core consisting of 69 fuel assemblies. Since the fuel burnup in the subcritical mode was limited to $\approx 40\,\mathrm{MW\,d\,t_{HM}^{-1}}$ in 450 EFPD, the fuel cycle was increased up to 6×90 EFPD. Furthermore, the total number of fuel assemblies in the subcritical core was increased to 72 and a mixed shuffle strategy was chosen to slightly reduce the peaking factors (enhanced in the ADS mode) and to heighten as much as possible the neutron flux performances (Sarotto, Fernandez, et al., 2012).

The results of the equilibrium cycle analysis of both cores are summarized in Table 2.4. The estimated total number of fresh fuel assemblies to be loaded in the reactor at the beginning of each new cycle is 15 for the critical mode and 12 for the subcritical mode.

In-Core Nuclear Fuel Management

In case of MYRRHA, in-core fuel management reduces to making decision on the loading pattern design. The other fuel management problems (Section 2.1.2) are irrelevant in this case, since only one type of a fuel assembly, characterized by a uniform fuel enrichment among the fuel pins and the absence of burnable poisons, will be used during the operation.

The MYRRHA loading pattern optimization problem has to be solved for every upcoming fuel cycle in order to reflect actual changes in the initial conditions and fuel cycle objectives, which are generally different from those assumed for the out-of-core analysis. Although no formal guidelines have been formulated for the reactor operation up to the present day, it is reasonable to assume two families of optimization objectives for MYRRHA: objectives aimed at the maximum irradiation performance of the facility (e.g., maximum neutron fluence in certain positions in the reactor core) and economically motivated objectives (e.g., maximum fuel discharge burnup or maximum BOC or EOC multiplication factor).

In-core fuel management optimization constraints are usually derived from the nuclear safety and reactor core operating limits. Because these limits have not been established for MYRRHA yet, a list of in-core fuel management optimization constraints must be first specified. Recalling Section 2.1.2, this list usually includes constraints on fuel cycle length (90 EFPD) and QOI production (e.g., minimum neutron flux levels), reactivity limits (e.g., EOC criticality condition for the critical core, shutdown margins), thermal limits (maximum cladding temperature limit), and additional constraints obtained on the output

from the out-of-core analysis (total number of fuel assemblies, number of fresh fuel assemblies, arrangement of the reactor shutdown systems, etc.)

The remainder of the dissertation deals with various MYRRHA loading pattern optimization problems solved by optimization methods using reactor analysis codes (Section 2.2.4)—the most suitable methods for a multipurpose flexible irradiation facility like MYRRHA. Chapter 3 is dedicated to the MYRRHA reactor models used by the optimization methods described afterward in Chapter 4. The software implementation of the solution scheme is then described in Chapter 5.

2.5 Summary

The second chapter of the dissertation gives general information on the areas directly related to the central topic of the work, a good grasp of which is necessary to make a coherent and justifiable selection of problem objectives and constraints, modeling methods and analytical tools, and techniques used to solve the MYRRHA loading pattern optimization problem.

The first part of the chapter deals with nuclear fuel management, a complex multidisciplinary decision making process aimed at safe and economically sound exploitation of a nuclear facility. The global fuel management problem of minimizing the fuel cycle cost, while satisfying all safety and operational constraints, is generally approached by breaking it up into the out-of-core and in-core subproblems. These two tightly coupled problems with different objectives, constraints, and decision variables are solved separately for practical reasons, employing core reload design methods tailored to different levels of simplification (scoping, preliminary, final methods). In-core fuel management involves making decisions on the fuel assembly loading pattern design and, if appropriate, burnable poison placement and control rod pattern design. At this stage, the fuel cycle cost is reduced by optimizing various surrogate parameters such as BOC k_{eff} or fuel discharge burnup, while staying within the reactivity limits, thermal margins, and other constraints given by the operational requirements and results of the out-of-core analysis.

The in-core fuel management by its definition postulates an optimization problem. Solution of this highly complex problem has been first done heuristically, depending solely on trial and error methods. This has led to several core loading designs (e.g., the low-leakage design) widely used in the nuclear industry up to the present day. However, developments in computer technology and modeling capabilities have allowed for a more rigorous approach to the solution of the problem, relying on conventional mathematical

programming methods using simplifying mathematical models or on methods based on some sort of an intelligent search that use ordinary reactor analysis codes. The latter group of optimization methods, most often represented by metaheuristic optimization algorithms, has received a notable attention in the nuclear community, being implemented in several production codes.

Metaheuristics are general purpose optimization methods proven to be effective in tackling difficult combinatorial problems with an immensely large number of possible solutions such as the loading pattern optimization problem. In the methods like GA or ACO, the search space is explored following rather simple rules in a hope that the evaluated trial solutions progressively evolve from random ones toward better ones. One of the biggest advantages of metaheuristics is their inherent flexibility, the ability to cope with arbitrary objectives and constraints, typically evaluated by a black-box solver. In loading pattern optimization, this black-box solver is represented by a reactor analysis code. In this way, any reactor system can be optimized using proper modeling methods and tools. On the other hand, the use of advanced modeling techniques in a combination with a large number of evaluated designs entails a substantial computational burden, the most significant drawback of this optimization approach.

Since the quality of an optimal solution can be only as good as the quality of the neutronics model used to derive this solution, it is very important to use adequate modeling tools and to balance properly their modeling accuracy with their computational requirements. The deterministic neutronics analysis codes used for reactor loading pattern optimization calculate basic core characteristics as the neutron flux and power distributions and multiplication factor. These codes solve numerically the steady-state neutron transport or diffusion equation in a step-wise manner, utilizing many simplifying assumptions generally different for different reactor technologies such as fast/thermal spectrum reactors or critical/subcritical accelerator driven systems. The correct use of the neutronics analysis codes thus requires a competent user aware of their limitations.

The last part of the chapter describes MYRRHA, a flexible material testing reactor under development at SCK•CEN in Mol in Belgium. MYRRHA is a fast-spectrum lead-bismuth-cooled reactor of 100 MW maximum power output able to operate in both critical and subcritical modes. The special nature and purpose of the machine imply special objectives and constraints to be considered within the MYRRHA in-core fuel management. In view of the preceding sections, the maximum irradiation performance and maximum multiplication factor are identified as the MYRRHA loading pattern optimization objectives, given a stock of available fuel assemblies and thermal limits represented by the maximum fuel cladding temperature.

Chapter 3

MYRRHA Reactor Core Modeling

The loading pattern optimization problem solution approach adopted in this work is based on optimization techniques that use reactor analysis codes to evaluate trial loading pattern designs. Since many loading patterns have to be evaluated during each optimization run (typically from tens of thousands to hundreds of thousands of evaluations), a trade-off between the model detail and computational load has to be found that allows to deliver a good-quality solution within a reasonable time frame[1].

As indicated in Chapter 2, the basic loading pattern characteristics that need to be calculated at the preliminary design stage, when the loading pattern optimization takes place, are the multiplication factor k_{eff} and the power and neutron flux distributions. The models and computational tools used to evaluate these quantities are usually different for different types of reactor systems, depending on their neutron spectrum, purpose, fuel design, etc. The special character of MYRRHA (fast neutron spectrum, single fresh fuel enrichment, absence of BPs) allows to employ many assumptions that simplify the whole core reload analysis process. For instance, the depletion calculation can be omitted in some cases and the key characteristics can be evaluated at BOC only, as k_{eff} can be approximated by a decreasing linear function of burnup. Another supporting argument for that is that the most severe conditions with respect to the thermal limits occur at BOC, when the largest power peaking factors are achieved. Moreover, when optimizing the MYRRHA-FASTEF critical reactor

[1] A reasonable time for performing loading pattern optimization for MYRRHA is in order of days, ideally in order of hours (e.g., overnight optimization).

core (see Figure 2.4) with a shutdown system that has anti-reactivity worth well above the required levels (cf. Section 2.4.4, Table 2.5), it can be assumed that the performance of the shutdown system will be satisfactory for each loading pattern and no additional calculations need to be performed at this stage.

The simplified scheme used for MYRRHA core reload analysis is shown in Figure 3.1. As the picture shows, only BOC calculation ($I = 0$) or BOC and EOC calculations ($I = 1$) are assumed in the simplified scheme, depending on the analysis objective, the thermal-hydraulics and control system feedback effects are disregarded, and the thermal-hydraulics analysis is limited to calculation of the maximum cladding temperature T_{clad}, the most restrictive thermal limit factor for MYRRHA.

In this chapter, the reactor analysis tools and models are described that are used in Chapter 4 for MYRRHA-FASTEF loading pattern optimization. Most of the attention is given to neutronics codes for k_{eff} and neutron flux and power distribution calculations. The analytical model that calculates T_{clad} as a function of the number of fuel assemblies N_{FA}, the core power fraction p_{f}, and the radial (P_{r}) and axial (P_{z}) power-peaking factors,

$$T_{\mathrm{clad}} = f(N_{\mathrm{FA}}, p_{\mathrm{f}}, P_{\mathrm{r}}, P_{\mathrm{z}}) \ , \tag{3.1}$$

is described briefly in Appendix B.

3.1 Neutronics Codes

Several computing tools have been developed for fast reactor neutronics analysis, including dedicated tools for cross section generation, whole-core analysis, and depletion calculation. Some of these tools have been adopted and used within the SCK•CEN Nuclear Systems Physics expert group for studying fast spectrum systems, among them the DIF3D whole-core neutronics code (see Section 3.1.2). It was decided at the beginning of the PhD project that this particular code will be used for MYRRHA core loading optimization for the following reasons: it is a well documented and maintained software specially developed for the needs of fast reactor research; as such, it complies with all requirements on an analysis code for MYRRHA loading pattern optimization (e.g., hexagonal geometry, computationally inexpensive diffusion and transport nodal solvers, capability to solve a fixed source problem); the code is readily available for the Organization for Economic Co-operation and Development (OECD) countries via the Nuclear Energy Agency (NEA) Data Bank computer program service (OECD-NEA, 2014); furthermore, DIF3D models of MYRRHA XT-ADS designs were available at the beginning of the PhD project (Van den Eynde, 2003) and

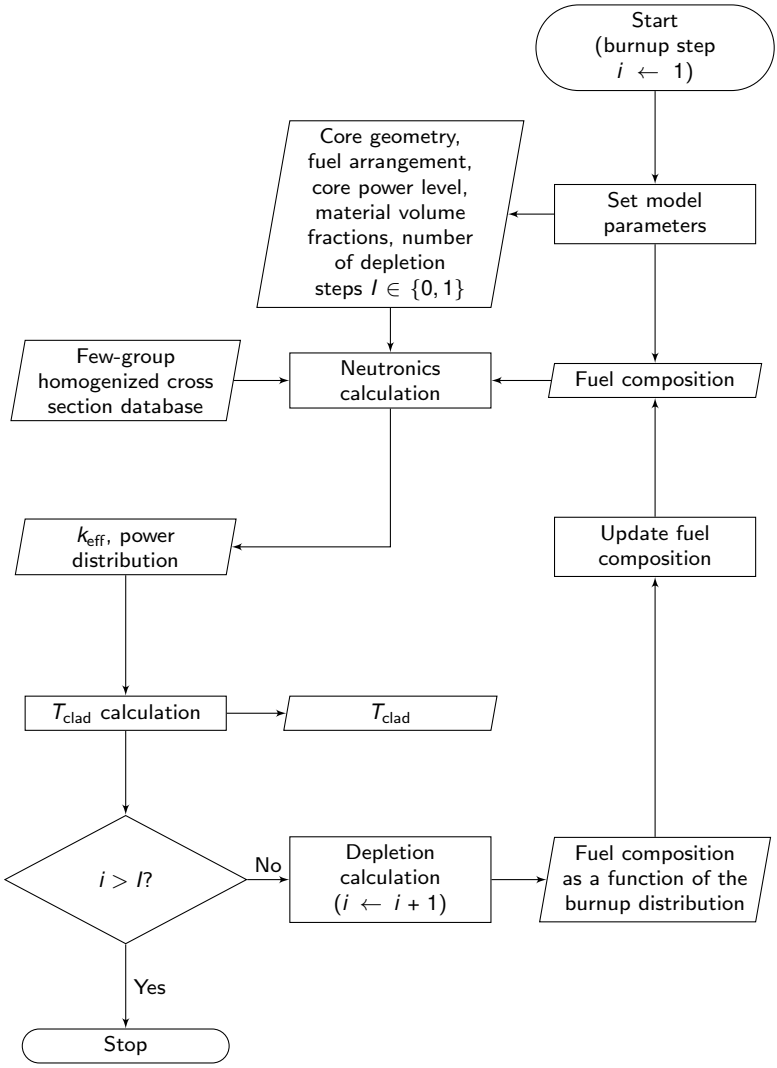


Figure 3.1: Simplified computational scheme for MYRRHA core reload analysis

could be used as a starting point for the development of MYRRHA-FASTEF models; and also, the PhD student had previous work experience with the code.

The DIF3D code is a part of a broader family of fast reactor analysis tools developed at the Argonne National Laboratory (ANL) in the US, some of which were also adopted in this work. It is the case of the MC^2 code for few-group cross section generation and the REBUS code for the depletion and fuel cycle analysis. All three codes are described in the following sections 3.1.1–3.1.3.

3.1.1 MC^2

The MC^2 code (ANL, 2014c; Henryson II et al., 1976) employs the ultra-fine group method, which is, besides the Bondarenko self-shielding factor method and the subgroup method, one of the main approaches used for fast reactor cross section generation (W. S. Yang and Taiwo, 2004). The ultra-fine group method was originally intended specifically for the treatment of resonance structures of the structural and metal coolant materials present in a fast reactor core (C. H. Lee and W. S. Yang, 2007). It is based on detailed spectrum calculation for individual compositions, allowing for a more rigorous energy treatment.

Code Features and Use

MC^2 calculations provide composition and temperature dependent cross sections in a user specified energy group structure and numerous other capabilities such as isotope mixing, delayed neutron data processing, free-format input, and flexibility in output data selection. Broad-group cross sections for whole-core calculations can be obtained directly from MC^2 calculations in the Committee on Computer Code Coordination (CCCC) ISOTXS format required on the input to the DIF3D code.

In this work, the MC^2-2 version of the code prepared in the year 2000 was used[2]. It contains cross section libraries based on the ENDF/B-V nuclear data file (Kinsey, 1983).

[2]A new improved version of the code, MC^2-3, was released in 2012 via the Radiation Safety Information Computational Center (RSICC) websites (ORNL-RSICC, 2014) under restricted distribution conditions. This version could not be used for this work because of these restrictions.

Method of Solution

The MC^2 code employs the ultra-fine group method to solve the slowing-down equation for specific compositions and temperatures with explicit representation of resonances. It solves the extended transport P_1, extended transport B_1, consistent P_1, and consistent B_1 fundamental mode ultra-fine group equations. Resolved and unresolved resonances are treated explicitly by the generalized resonance integral formulation based on the narrow resonance approximation, including overlapping and Doppler broadening effects. A fundamental mode homogeneous unit cell calculation is performed by solving the multi-group slowing-down equation above the resolved resonance energy and the continuous slowing-down equation below this range. Equivalence theory is used to treat the heterogeneity effect, and isotropic approximation is used for fission, inelastic, and (n,2n) sources (W. S. Yang, 2012).

Supporting Data Libraries

The multi-group cross-section libraries utilized by MC^2 are prepared by the ETOE-2 code (ANL, 2014b), by processing the ENDF/B data files. These libraries include resolved resonance parameters, unresolved resonance parameters, ultra-fine-group smooth cross sections (2082 groups for energies between 0.41 eV and 14.19 MeV), inelastic and (n,2n) scattering data, fission spectrum parameters, and elastic scattering distributions.

3.1.2 DIF3D

DIF3D (ANL, 2014a) is a deterministic neutronics code specifically designed for fast reactor whole-core analysis. It solves the steady-state multi-group diffusion and transport equations in two- and three-dimensional hexagonal and Cartesian geometries using various solvers.

Code Features and Use

DIF3D solves the eigenvalue, adjoint, fixed source, and criticality (concentration) search problems. Flux and power density maps by mesh cell or node and region balance integrals are provided. Although primarily designed for fast reactor problems, upscattering is also treated (ANL, 2014a). Likewise MC^2, DIF3D also allows for free-format input and provides a large flexibility in output data selection. For instance, the neutron flux and power density maps can be printed

in a simple format that visually represents the core geometry on a mesh/region level, using ASCII characters only.

The revised DIF3D10.0 release from 2011 was used for this work.

Method of Solution

The DIF3D code provides three flux solution options: finite-difference diffusion theory (Derstine, 1984), nodal diffusion theory (Lawrence, 1983), and variational nodal transport theory (Lewis and Miller, 1993; Palmiotti et al., 1995) methods. The coarse-mesh nodal methods are considered to be the most suitable methods for fuel management studies as well as for loading pattern optimization due to their enhanced accuracy over the finite-difference option and a potential order-of-magnitude reduction in the computation cost of a three-dimensional calculation (Lawrence, 1983).

Nodal diffusion method. The DIF3D nodal option solves the multi-group steady-state neutron diffusion and (for Cartesian geometry only) transport equations using a nodal scheme with one mesh cell (node) per hexagonal assembly or one Cartesian geometry cell specified by a user. The nodal equations are derived using higher order polynomial approximations to the spatial dependence of the flux within the node. The final equations involve spatial moments of the node-interior flux distribution plus surface-averaged partial currents across the faces of the node.

Variational nodal transport method—VARIANT. A short description of the variational nodal transport option is given by W. S. Yang (2012): The VARIANT code (Palmiotti et al., 1995) is a typical production code based on the second-order formulation. It is based upon a variational nodal method that guarantees nodal balance and permits refinement using hierarchical complete polynomial trial functions in space and spherical harmonics or simplified spherical harmonics in angle. Flux and source expansions of up to sixth order and partial current expansions up to second order are allowed. Angular and scattering expansions of up to P_5 are allowed. The multi-group even angular parity flux equations are solved within the nodes by reformulating it as a variational principle.

VARIANT provides highly accurate transport theory results at a fraction of the computing cost of alternative transport methods like discrete ordinates or Monte Carlo methods (Waltar et al., 2012).

3.1.3 REBUS

REBUS (ANL, 2014d; Hosteny, 1978; Toppel, 1990) is a versatile code system for carrying out fast-reactor burnup and fuel cycle analysis calculations. It models nuclide transmutations on a three-dimensional, region-dependent basis by making use of DIF3D as a flux calculation module.

Code Features and Use

Two basic types of problems are solved by the REBUS code: an equilibrium (or infinite-time) cycle problem to determine the equilibrium condition of a reactor operating under a fixed periodically repeating fuel management scheme and a non-equilibrium cycle problem to determine the explicit cycle-by-cycle condition of a reactor under a specified periodic or non-periodic fuel management scheme. REBUS provides considerable flexibility for specifying operational constraints and fuel management strategies for both in-core and out-of-core portions of the fuel cycle. Search options for fresh fuel enrichment, control poison density, or reactor burn cycle time are available, allowing the user to achieve a specified multiplication factor or discharge burnup without time consuming (trial and error) repetitions of the analysis (W. S. Yang, 2012).

The code allows flexible user-defined burnup chains, with no limit on the number of nuclides. The isotopes to be considered in the burnup equations, as well as their transmutation reactions, are specified by the user. Ten reaction types are permitted. These are (n,f), (n,γ), (n,p), (n,α), (n,2n), (n,d), (n,t), β^--decay, β^+-decay, and α-decay. Other data required for depletion calculation like the nuclide decay constants or the energy released per fission can also be supplied as input parameters.

The results presented in the following chapter were obtained using REBUS-PC 1.4 (Olson, 2001).

Method of Solution

The total reactor burn cycle time is divided into one or more subintervals, the number of which is specified by the user. An explicit nuclide depletion computation is performed in each region of the reactor over each of these subintervals using the average reaction rates over the subinterval. These average reaction rates are based on fluxes obtained from an explicit one-, two-, or three-dimensional diffusion or transport theory neutronics solution computed at both the beginning and end of the subinterval (ANL, 2014d).

The depletion equations (2.40) are solved by the matrix-exponential technique.

3.2 Neutronics Models

The MYRRHA-FASTEF neutronics models used in Chapter 4 for MYRRHA core loading optimization are described in this chapter, including models for few-group cross section generation, whole-core analysis, and depletion analysis. All models were developed primarily for the critical variant of the MYRRHA-FASTEF design, assuming the use of the neutronics codes described in the preceding section. Several computational options and model parameters and approximations are considered and discussed, where these significantly influence the computational time and/or the model accuracy—the two characteristics essential for successful and efficient loading pattern optimization. The data concerning the MYRRHA-FASTEF core geometry and material compositions were mostly taken from Sarotto, 2011.

3.2.1 Few-Group Cross Sections Generation

Two sets of data have to be provided on the input to the MC^2 code for few-group cross sections generation. The first set of data describes the physical design and other properties of the fuel lattice of the modeled system, that is, its geometry, constituent materials (i.e., mixtures of nuclides), working temperatures, etc. Other nuclides like the fission products and higher actinides considered in the depletion problem, are also included here. The second set of data consists of various parameters used by the code internally to produce the required results (e.g., opted broad-group structure) using chosen solvers and execution options.

Model Description

Fuel geometry. The lattice cell calculation is performed for homogenized fuel assembly, meaning that all heterogeneity effects within the fuel assembly are neglected. This is a simplifying assumption often used in fast reactor analysis.

Material composition and other properties. The fuel composition used in the lattice cell calculation was obtained based on the atomic densities of the nuclides present in the constituent materials and their corresponding volume fractions. Both materials and volume fractions are shown in Table 3.1. The nuclide compositions of all materials except from the MOX fuel can be found in

Table 3.1: MYRRHA-FASTEF fuel cell composition

Component	Material	Vol. fraction (%)
Fuel column[a]	MOX	30.74
Filling gas	^{4}He (5 bar)	2.93
Pin cladding[b]	Ti-Ti15 SS	14.81
Assembly wrap	T91 FMS	7.29
Coolant	LBE	44.22

[a] The central (stiffness) pin does not contain fuel.
[b] Includes the helical spacer wire made of the same material.
Source: Sarotto, 2011.

Appendix C. The volume fraction values in Table 3.1 were obtained based on a detailed fuel cell geometry (e.g., Sarotto, 2011). The fresh fuel composition[3] and fuel cell geometry cannot be presented here due to the confidentiality reasons.

Besides the nuclides present in the fuel cell, which are used for calculation of the problem specific ultra-fine-group weighting spectrum, all other nuclides considered in the reactor neutronics analysis are also specified in the input. Higher actinides and fission products contained in depleted fuel as well as nuclides contained in structural materials, absorbers, reflectors, etc. are included in zero atomic densities (infinitely diluted), so their few-group cross sections can be generated without having any influence on the weighting spectrum, included in the ISOTXS binary file, and used consecutively for solving the whole-core or depletion problems.

In order to simplify the whole-core analysis, the MC^2 mixing module was used to generate the cross sections of one fictitious lumped fission product (LFP) that represents 100 most important individual fission products abundant in MYRRHA-FASTEF fuel—see Appendix D.

The temperature of the fuel meat was set to 1200 °C and the temperature of all other components was set to 360 °C. The thermal energy yield per fission values[4] used in the model are shown in Table 3.2.

[3] The MYRRHA-FASTEF fresh fuel is made of natural U and 34.5 wt% of Pu with a small fraction of Am (< 2 wt% of the Pu content). The applied Pu isotopic vector corresponds to the one representative for PWR spent fuel of 45 GW d t_{HM}^{-1} burnup, with a 4.5 wt% initial ^{235}U enrichment, and after 50 years of cooling (Sarotto, 2011).

[4] Dependence on incident neutron energy was neglected.

Table 3.2: Thermal energy yield per fission (ENDF/B-VII.1)

Actinide	En. yield (MeV)	Actinide	En. yield (MeV)	Actinide	En. yield (MeV)
^{234}U	191.84	^{238}Pu	200.36	^{242m}Am	205.72
^{235}U	193.41	^{239}Pu	198.90	^{243}Am	203.62
^{236}U	194.49	^{240}Pu	199.47	^{242}Cm	203.46
^{237}U	187.82	^{241}Pu	201.98	^{243}Cm	204.02
^{238}U	197.79	^{242}Pu	202.78	^{244}Cm	208.38
^{237}Np	196.37	^{241}Am	201.96	^{245}Cm	205.22

Solution Parameters

Only one weighting spectrum calculated for fresh fuel was used for energy group condensation. Few-group cross sections obtained based on the same weighting spectrum were then used also for a depleted core. It is possible to make this assumption as the neutron spectrum in fast reactors changes slowly with burnup. Furthermore, no spatial dependence of the cross sections was assumed in the whole-core model; that is, the same region-independent cross-section constants were considered in every part of the modeled system[5].

33 and 15 group structures form Table 3.3 were opted for the condensation from the 2082 fine energy groups.

The ultra-fine method was used in MC^2-2 for solving the consistent P_1 problem. Default values for other computational options were used. The ultra-fine and broad-group neutron spectra resulting from the MC^2-2 calculations are depicted in Figure 3.2 and compared with the reference neutron spectrum coming from the MCNP results.

Nuclear constants to support linearly anisotropic scattering treatment in the whole-core nodal transport calculations were generated and included in the ISOTXS file.

[5]In order to verify this assumption, the same MYRRHA-FASTEF reactor core was first evaluated with the SCALE 6.1 KENO-6 Monte Carlo code (ORNL, 2011) using two different sets of cross-section data with a minimum number of energy groups and the obtained results were then compared. While no spatial dependence was assumed in the first cross-section data set, four radial regions were assumed for the generation of the second set (two fuel regions, reflector, and outer-core region). Both cross-section sets were prepared using the SCALE 6.1 CSAS sequence (ORNL, 2011). It was concluded that the observed differences in the results were negligible (Mylonakis, 2012).

Table 3.3: Energy group structures

	33 groups		15 groups	
Gr.	E (eV)	Δu	E (eV)	Δu
1	1.96×10^{7}[a]	0.675	1.96×10^{7}[a]	1.175
2	1.00×10^{7}	0.500	6.07×10^{6}	1.000
3	6.07×10^{6}	0.500	2.23×10^{6}	0.500
4	3.68×10^{6}	0.500	1.35×10^{6}	1.000
5	2.23×10^{6}	0.500	4.98×10^{5}	1.000
6	1.35×10^{6}	0.500	1.83×10^{5}	1.000
7	8.21×10^{5}	0.500	6.74×10^{4}	1.000
8	4.98×10^{5}	0.500	2.48×10^{4}	1.000
9	3.02×10^{5}	0.500	9.12×10^{3}	1.500
10	1.83×10^{5}	0.500	2.03×10^{3}	1.500
11	1.11×10^{5}	0.500	4.54×10^{2}	3.000
12	6.74×10^{4}	0.500	2.26×10^{1}	1.732
13	4.09×10^{4}	0.500	4.00×10^{0}	2.002
14	2.48×10^{4}	0.500	5.40×10^{-1}	1.686
15	1.50×10^{4}	0.500	1.00×10^{-1}	6.812
16	9.12×10^{3}	0.500		
17	5.53×10^{3}	0.500		
18	3.35×10^{3}	0.500		
19	2.03×10^{3}	0.500		
20	1.23×10^{3}	0.508		
21	7.42×10^{2}	0.492		
22	4.54×10^{2}	0.400		
23	3.04×10^{2}	0.717		
24	1.49×10^{2}	0.483		
25	9.17×10^{1}	0.300		
26	6.79×10^{1}	0.525		
27	4.02×10^{1}	0.575		
28	2.26×10^{1}	0.500		
29	1.37×10^{1}	0.500		
30	8.32×10^{0}	0.732		
31	4.00×10^{0}	2.002		
32	5.40×10^{-1}	1.686		
33	1.00×10^{-1}	6.812		

[a] 1.42×10^{7} for DIF3D and REBUS calculations. Upper energy-group boundaries are indicated. *Source:* Rimpault et al., 2002.

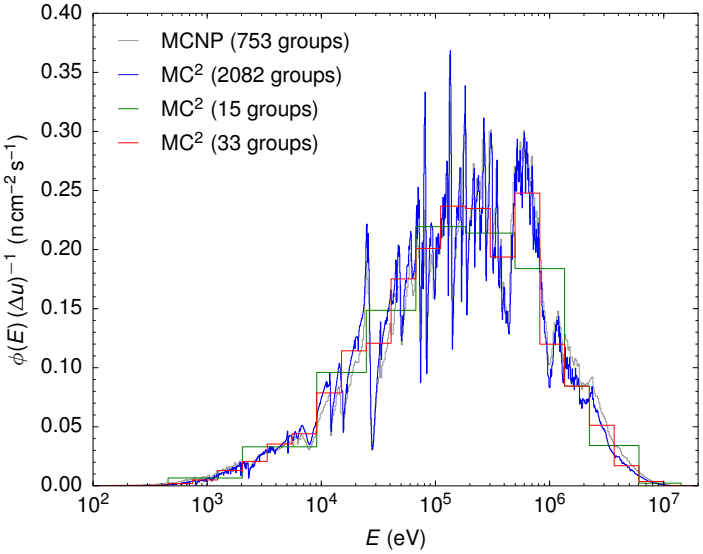


Figure 3.2: MYRRHA-FASTEF neutron spectrum (normalized to 1.0); MCNP results (inner fuel position): group-averaged mean values (*source:* Malambu, 2014); MC² results (infinite homogenized fresh fuel lattice)

3.2.2 Whole-Core Calculation

The MYRRHA-FASTEF whole-core neutronics analysis was performed using DIF3D10.0 supported by cross-section data in the ISOTXS format provided by MC²-2. The adopted whole-core model consists of the description of the three-dimensional core geometry and materials composed of the nuclides present in the ISOTXS file. The materials and material mixtures are then assigned to each geometrical unit (or region). The neutronics solver and the solution parameters are also specified in the DIF3D input file.

The neutronics model described below concerns only the critical MYRRHA-FASTEF design optimized in Chapter 4. A similar model can be developed for the subcritical variant with a slightly modified core configuration (cf. Figure 2.4—absence of SRs, increased number of FAs, presence of the central spallation SA). In order to solve the heterogeneous problem (2.20), the model must contain information on the spatial distribution of the external neutron source densities per each energy group.

Model Description

Core geometry. The MYRRHA-FASTEF three-dimensional geometrical model is composed of hexagonal-z prisms (units, cells, components) organized in a regular hexagonal grid. That means that every technological component has to be approximated by such a hexagonal cell, including the cylindrical core barrel and the coolant volume outside of the barrel (see Figure 3.3), which are considered in the model in order to provide more realistic boundary conditions for the system characterized by a longer mean free path and increased neutron leakage.

The model exhibits one-third rotational symmetry (see Figure 3.3). The geometry was adopted at cold (20 °C) conditions.

Core components and materials. In Figure 3.4, all cell types from Figure 3.3 are divided into several axial zones made of homogenized material mixtures that represent their different functional parts:

LBE	The reactor coolant.
NOZZLE	A T91 stainless steel cone that serves as an inlet or outlet structure for LBE in the fuel assemblies or dummy assemblies.
LBE_WRAP	A hollow T91 cylinder between the nozzle and the fuel rod grid.
GRID_PIN	A fuel assembly part that contains the end points of the fuel rod grid.
PLENAINF	The fuel assembly rods below the part with the actual fuel, filled with He.
YZRO_INSU	The top and bottom YZrO fuel insulator parts.
FUEL_i	The fuel assembly part that contains the actual fuel of age i.
PLENASUP	The fuel rods above the part with the actual fuel, filled with He.
TOPPLATE	The T91 top plate of the entire fuel assembly, which holds the assmebly into place.
HE_IPS	A small section in the IPS between the empty and the full parts of the assembly, filled with He.
YZRO_IPS	The YZrO insulator in the experimental assembly.
SAMPLE_IPS	A part containing experimental samples to be irradiated. In this case, this part of the assembly is filled with Al alloyed with Mg.

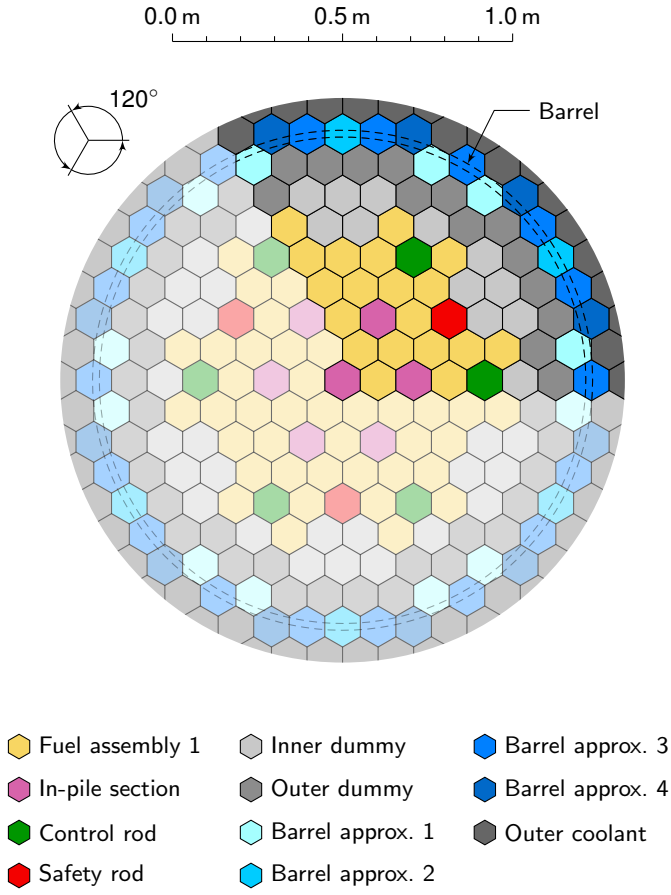


Figure 3.3: MYRRHA-FASTEF BOL core model

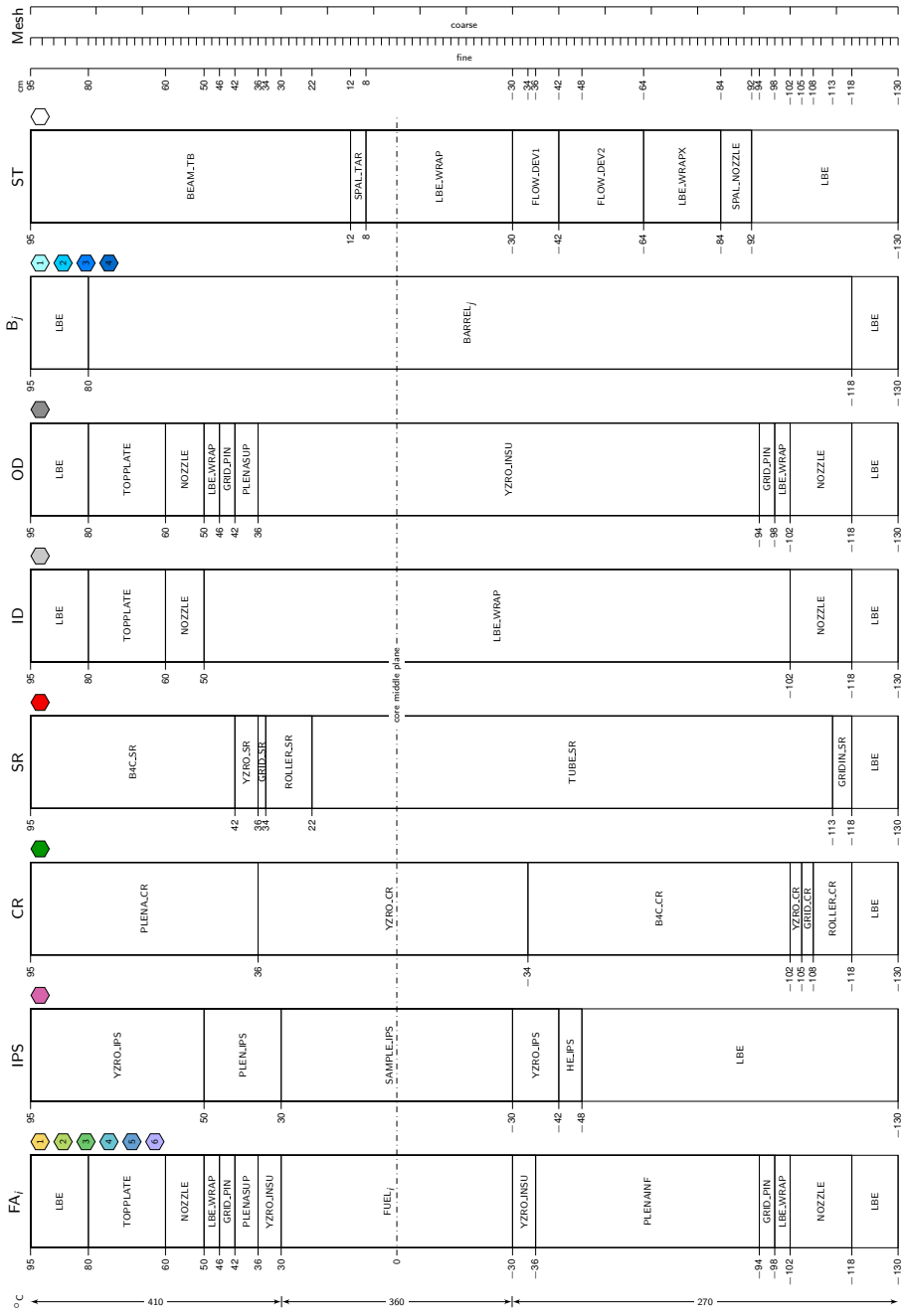


Figure 3.4: MYRRHA-FASTEF subassemblies models

PLEN_IPS	The IPS plenum.
ROLLER_CR	The machinery which allows the control rod to move within its assembly case.
GRID_CR	The stainless steel control rod grid.
YZRO_CR	The YZrO control rod insulator.
B4C_CR	The boron alloy that serves as the neutron absorber.
PLENA_CR	The control rod plenum.
GRIDIN_SR	The lower grid of the safety rod assembly, not attached to the safety rod itself.
TUBE_SR	A hollow tube while the safety rod is not inserted.
ROLLER_SR	The machinery that allows the safety rod to move within its tube.
GRID_SR	The part of the grid in the safety rod assembly connected to the safety rod.
YZRO_SR	The safety rod insulator.
B4C_SR	The boron-filled part which performs the main function of the safety rod.
BARREL$_i$	The hexagonal approximations of the core barrel stainless steel cylinder. The index i denotes different steel-to-LBE volume-fraction ratios.
SPAL_NOZZLE	The stainless steel spallation assembly inlet nozzle.
LBE_WRAPX	A modified assembly wrap.
FLOW_DEV1	The first section of the spallation assembly part containing the device for controlling the LBE flow through the assembly.
FLOW_DEV2	The second section of the spallation assembly flow device part.
SPAL_TAR	The part with the beam window and spallation target.
BEAM_TB	The accelerator beam tube.

The compositions of all material mixtures are specified in Appendix C in Table C.9. Depending on the applied axial mesh (see the right part of Figure 3.4), the mixtures can be mixed further in the model, so each mesh cell is filled only by a single mixture. The densities of solid materials were obtained assuming cold conditions[6]. Three temperature zones were considered for the coolant density (see the left part of Figure 3.4): 270 °C for the region below the core (average inlet temperature), 360 °C for the fuel region, and 410 °C for the region above the core (average outlet temperature).

[6]It should be noted that the combined effect of the cold geometry and densities of the solid materials yields ≈ 500 pcm k_{eff} overestimation for the MYRRHA-FASTEF neutronics models (Sarotto, 2011).

Solution Parameters

Two steady-state DIF3D solvers were used in the analysis: the VARIANT nodal transport solver and the nodal diffusion solver, simply referred to as DIF3D-diffusion hereinafter.

The coarse axial mesh, ten rings of the outer-coolant cells, and 20 cm of LBE above and below the core model were selected for the transport calculations with the zero incoming angular flux ($\psi^{-}_{\partial V}$) outer-boundary conditions. The following VARIANT solver options were applied: 240/-33/1[7]. The values of these parameters/options were obtained based on the results of a conducted parametric study. Default values were used for other solver parameters.

The nodal diffusion calculations were performed with the 4^{th} order of the polynomial approximation to the one-dimensional fluxes in the xy-plane and the 0^{th} order of the polynomial approximation to the leakages transverse to the x- and y-directions. The cubic (3^{rd} order) flux approximation and the quadratic (2^{nd} order) leakage approximation were adopted for the z-direction[8]. Again, default values were used for other parameters. Application of the albedo external boundary conditions (β) in the xy-plane was tested for the diffusion option, as discussed further in Section 3.3.1.

The same eigenvalue convergence criterion 1.0×10^{-6} was used for all calculations.

3.2.3 Depletion Calculation

The MYRRHA-FASTEF depletion analysis was performed using the REBUS code with the steady-state models outlined in the preceding part on the whole-core calculations (Section 3.2.2). Besides the static reactor model, REBUS also requires a depletion model that includes description of transmutation reactions of interest between considered nuclides, various fission yield and decay constants, and REBUS execution parameters.

[7] DIF3D-VARIANT uses a triplet LMN (240 in this case) to describe the nodal spatial approximation, where $L \cdot \cdot$, $\cdot M \cdot$, $\cdot \cdot N$ are the orders of the polynomial approximations representing the neutron sources withing the nodes, neutron fluxes within the nodes, and neutron leakages on the surfaces of the nodes, respectively. A duet MN (-33 in this case) is used to represent the angular approximation, where $M \cdot$ and $\cdot N$ are, respectively, the orders of the P_N expansion of the flux and leakage. When these two values are negative, a simplified spherical harmonics approximation is used (Palmiotti et al., 1995). The last number (3 in this case) is the applied anisotropic scattering order L.

[8] Default values for the hexagonal geometry.

Table 3.4: Actinide decay constants (ENDF/B-VII.1)

Actinide	Decay	λ (s^{-1})	$T_{1/2}$ (s)
^{237}U	β^-	1.1885×10^{-6}	5.8320×10^{5}
^{238}Pu	α	2.5045×10^{-10}	2.7676×10^{9}
^{241}Pu	β^-	1.5371×10^{-9}	4.5096×10^{8}
^{242}Cm	α	4.9236×10^{-8}	1.4078×10^{7}
^{243}Cm	α	7.5479×10^{-10}	9.1833×10^{8}
^{244}Cm	α	1.2128×10^{-9}	1.9180×10^{8}

Model Description

The depletion calculations were performed taking into account nuclear transformations for nuclides ranging from ^{234}U to ^{245}Cm. One lumped fission product (LFP) was used to represent the fission products. The methodology used to generate the LFP cross sections is described in detail in Appendix D. The number of LFPs released per a fission of a given nuclide was calculated as a fraction of the atomic mass of the nuclide and the LFP (see Table 3.5). A dummy nuclide was also used to represent the other end products not included in the chains.

(n,γ), (n,2n), and (n,f) reactions were considered for all actinide isotopes included in the problem specification. In the capture and (n,2n) reactions, short-lived intermediate products were neglected. As a result, the products of capture reactions of ^{238}U, ^{237}Np, ^{242}Pu, and ^{243}Am were represented by ^{239}Pu, ^{238}Pu, ^{243}Am, and ^{244}Cm, respectively. The capture reaction of ^{241}Am was modeled to yield ^{242}Cm, ^{242m}Am, and ^{242}Pu with yield fractions of 0.68, 0.18, and 0.14, respectively. Ten most prominent (n,2n) reactions were considered in the depletion scheme. The end product of ^{245}Cm capture reaction was represented by a fictitious dummy isotope.

Important α- and β-decays of actinide nuclides were also considered. Specifically, α-decay was considered for ^{238}Pu, ^{242}Cm, ^{243}Cm, and ^{244}Cm. The β^--decays of ^{237}U, ^{241}Pu, and the β^+-decay of ^{242m}Am were also included in the burnup chains. All employed decay constants are overviewed in Table 3.4.

The resulting depletion scheme is depicted in Figure 3.5. It was obtained based on a thorough study of different depletion schemes used for fast-reactor analysis (e.g., Salvatores, 1986; Toshinsky et al., 1999; Tsvetkov et al., 2012), one-group macroscopic cross sections calculated by ALEPH (Stankovskiy, 2013), and actinide decay constants from ENDF/B-VII.1 (Chadwick et al., 2011). All

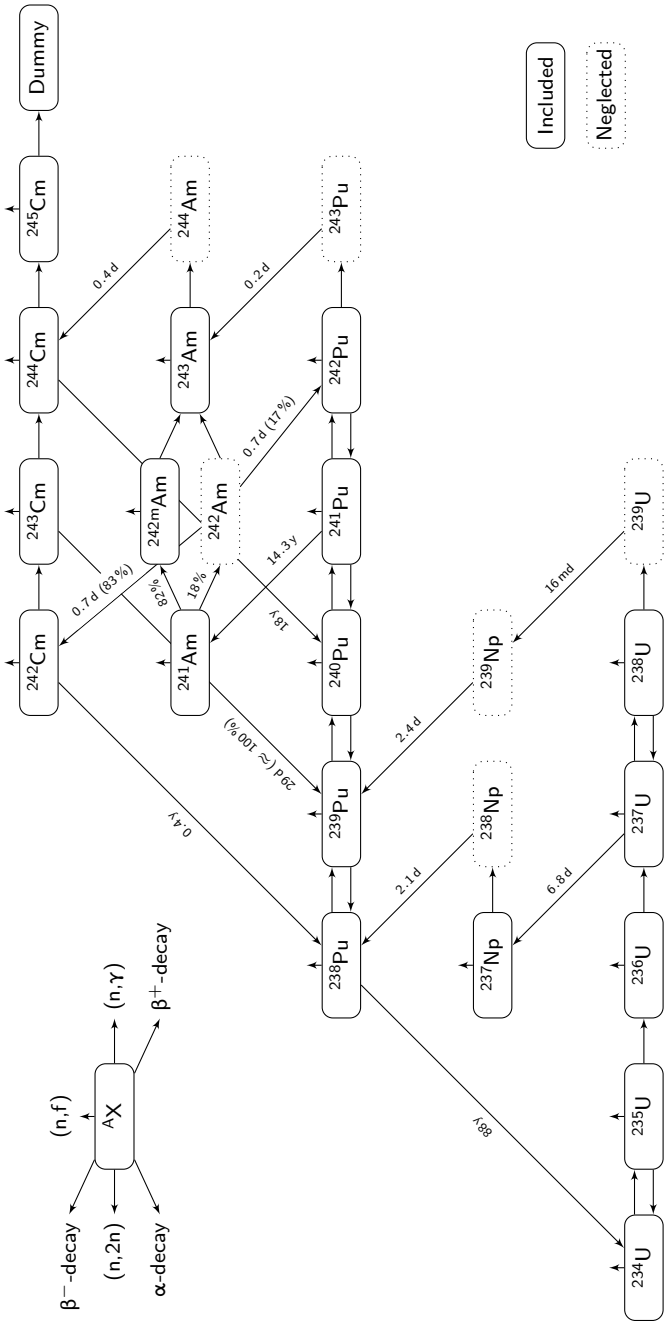


Figure 3.5: Actinide depletion scheme of interest

Table 3.5: Actinide transmutation reactions of interest

Initial actinide[a]	Reaction	Product actinide	Initial actinide[a]	Reaction	Product actinide
^{234}U	(n,f), 1.972	LFP	^{241}Pu	(n,γ)	^{242}Pu
^{234}U	(n,γ)	^{235}U	^{241}Pu	(n,2n)	^{240}Pu
^{235}U	(n,f), 1.980	LFP	^{241}Pu	β^--decay	^{241}Am
^{235}U	(n,γ)	^{236}U	^{242}Pu	(n,f), 2.039	LFP
^{235}U	(n,2n)	^{234}U	^{242}Pu	(n,γ)[b]	^{243}Am
^{236}U	(n,f), 1.989	LFP	^{242}Pu	(n,2n)	^{241}Pu
^{236}U	(n,γ)	^{237}U	^{241}Am	(n,f), 2.031	LFP
^{237}U	(n,f), 1.997	LFP	^{241}Am	(n,γ), 68 %[b]	^{242}Cm
^{237}U	(n,γ)	^{238}U	^{241}Am	(n,γ), 14 %[b]	^{242}Pu
^{237}U	β^--decay	^{237}N	^{241}Am	(n,γ), 18 %[b]	^{242m}Am
^{238}U	(n,f), 2.006	LFP	^{242m}Am	(n,f), 2.039	LFP
^{238}U	(n,γ)[b]	^{239}Pu	^{242m}Am	(n,γ)[b]	^{243}Am
^{238}U	(n,2n)	^{237}U	^{243}Am	(n,f), 2.048	LFP
^{237}Np	(n,f), 1.998	LFP	^{243}Am	(n,γ)	^{244}Cm
^{237}Np	(n,γ)[b]	^{238}Pu	^{242}Cm	(n,f), 2.039	LFP
^{238}Pu	(n,f), 2.006	LFP	^{242}Cm	(n,γ)	^{243}Cm
^{238}Pu	(n,γ)	^{239}Pu	^{242}Cm	α-decay	^{238}Pu
^{238}Pu	α-decay	^{234}U	^{243}Cm	(n,f), 2.048	LFP
^{239}Pu	(n,f), 2.014	LFP	^{243}Cm	(n,γ)	^{244}Cm
^{239}Pu	(n,γ)	^{240}Pu	^{243}Cm	α-decay	^{239}Pu
^{239}Pu	(n,2n)	^{238}Pu	^{244}Cm	(n,f), 2.056	LFP
^{240}Pu	(n,f), 2.023	LFP	^{244}Cm	(n,γ)	^{245}Cm
^{240}Pu	(n,γ)	^{241}Pu	^{244}Cm	α-decay	^{240}Pu
^{240}Pu	(n,2n)	^{239}Pu	^{245}Cm	(n,f), 2.065	LFP
^{241}Pu	(n,f), 2.031	LFP	^{245}Cm	(n,γ)	Dummy

[a] Active nuclides.
[b] Simplified reaction route (see Figure 3.5).
1.000 yield fractions and 100 % isomeric state branching fractions are assumed for all reactions unless explicitly stated otherwise.

transmutation reactions included in the model are summarized in Table 3.5.

Solution Parameters

The burnup interval spanned over the whole 90 days long MYRRHA fuel cycle period. Each fuel assembly was represented by one burn region and

the "region-density iteration" (Hosteny, 1978) was performed with a relative convergence criterion of 0.001. That is, the depletion calculation for each region was performed with the average of the beginning and end of time interval fluxes. The end of time interval flux was iteratively computed by iteration on the final nuclide densities. In this case of a relatively short fuel cycle and low fuel burnups, one region-density iteration was performed at most to satisfy this convergence criterion. The total burn cycle time consisted of one subinterval to minimize the computation time. Hence, spectrum calculations were performed at two "time nodes" only, at BOC and EOC.

3.3 Modeling Results

The above-described modeling approach was tested on two critical MYRRHA-FASTEF core configurations: the so-called beginning-of-life (BOL) core layout with 57 fresh fuel assemblies (Case 1), and the beginning-of-equilibrium-cycle (BOEC) core with 69 fuel assemblies described in Section 2.4.4 (Case 2). All control rods were fully withdrawn from the reactor core at BOC in both cases to simplify the analysis. As a consequence, the power peaking factors and the correspondent T_{clad} could result slightly underestimated[9]. The main calculated core characteristics were compared with those obtained using the MCNPX 2.7.E[10](/ALEPH 2.1[11]) or ECCO/ERANOS 2.0[12] codes.

3.3.1 Case 1

The BOL core layout is depicted in Figure 3.3. It was designed as a MYRRHA-FASTEF starting core filled with fresh fuel assemblies exclusively to allow to maintain the criticality throughout the whole standard 90 EFPD-long fuel cycle. Comparing to the BOEC core from Section 2.4.4, the reactor safety and control system remains the same as well as the number and positions of the IPS channels, even if, presumably, the first MYRRHA core will be loaded without them (Sarotto, 2011). The lower number of fuel assemblies was achieved by replacing 12 outer fuel assemblies with inner-dummy assemblies. The core was originally analyzed at the 82.6 MW power level (Sarotto, 2011) that should

[9] In general, the "worst" conditions for the cladding with highest peaking factors and fluxes are obtained for the BOC conditions, in which both the CRs partial insertion to compensate the fuel-cycle k_{eff} swing and the contribution of the in-out reshuffling scheme are taken into account (Sarotto, 2011).

[10] X-5 Monte Carlo Team, 2008, 2011.

[11] Stankovskiy and Van den Eynde, 2011; Van den Eynde, Stankovskiy, et al., 2013.

[12] Rimpault et al., 2002.

Table 3.6: MYRRHA-FASTEF BOL core characteristics (Case 1): results 1

	Neutronics code		
Characteristic	VARIANT	ERANOS	MCNPX
k_{eff}[a]	1.00200	1.00801	1.00942 ±0.00005
$q_{\max}$ (W cm^{-1})	310.4	302.2	
Power peaking			
Hot FA $P_{\text{r FA}}$	1.030[b]	1.022	1.026
Hot FA P_{z}	1.159	1.146	1.152
$P_{\text{r core}}$	1.392	1.346	1.361
P_{q}	1.662	1.576	1.608
Peak ϕ ($\text{n cm}^{-2}\,\text{s}^{-1}$)			
Central IPS	3.65×10^{15}	3.57×10^{15}	3.40×10^{15}
Satellite IPSs	3.18×10^{15}	3.13×10^{15}	2.98×10^{15}
T_{clad} (°C)	534.6		
Computation t	≈ 7.4 s[c]		≈ 13 h[13]

[a] The MCNPX result is in the $\mu \pm \sigma$ format.
[b] Set as a conservative estimate on the input to the calculation.
[c] Executed on a laptop with an Intel® Core™ i5 M520 processor (4×2.4 GHz) and 3.7 GiB of RAM.
Source: Sarotto, 2011 (ERANOS); Sarotto, Fernandez, et al., 2012 (MCNPX).

guarantee meeting the maximum fuel-cladding temperature limit discussed in Section 2.4.4.

This particular study focuses on BOC steady-state criticality analysis. Various results are presented for the VARIANT and DIF3D-diffusion solvers, including the estimation of k_{eff}, neutron flux and power density profile, power-peaking factors, maximum linear power, and maximum cladding temperature.

VARIANT calculations. The VARIANT results are summarized in Table 3.6 together with the results obtained using the MCNPX and ERANOS codes. The radial neutron flux distribution in the central part of the reactor is displayed in

[13] One typical steady-state MCNP/X calculation performed at NSP to evaluate MYRRHA takes around 13 hours when executed on the SCK•CEN Fermi computer cluster using eight Intel® Xeon® E5530 processors (64×2.4 GHz CPUs in total with hyper-threading) with sufficient RAM. Such calculation consists of a criticality run of 200 cycles with 1.0×10^6 particles per cycle to achieve a k_{eff} 1-σ error of 5.0×10^{-5}.

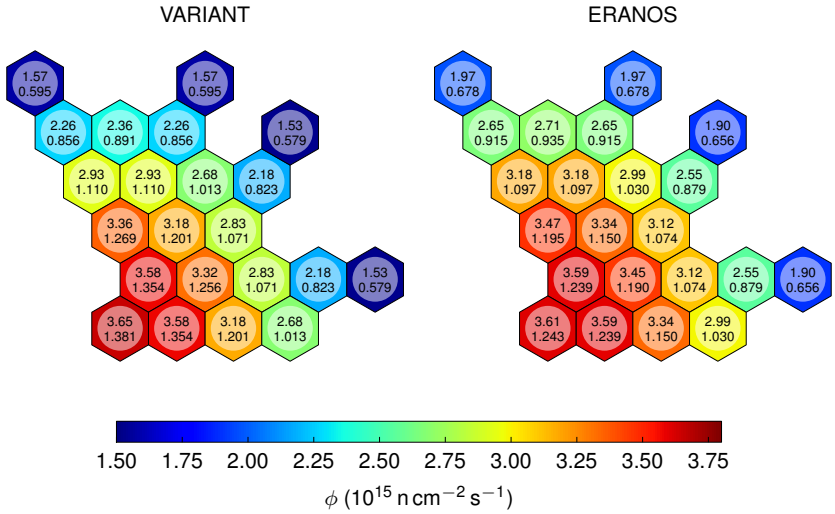


Figure 3.6: ϕ map (top values) and form factors (bottom values) for FA and IPS channels (Case 1): results 1; VARIANT results: average values for a 12 cm-thick central core region; ERANOS results: maximum values (*source:* Sarotto, 2011)

Figure 3.6[14] and the linear power axial distribution of the average pin in the hot fuel assembly $\bar{q}_{\text{hotFA}}$ is shown in Figure 3.7.

The largest discrepancy was observed in the k_{eff} values; the VARIANT calculation gives a value approximately 700 pcm lower than the reference MCNPX estimate. The peaking factors are generally larger, leading to conservative T_{clad} values. The peak neutron fluxes in the IPS sections are $\approx 7\,\%$ higher than the reference values but are very close to the ERANOS results. Also, the DIF3D and ERANOS curves overlap well in Figure 3.7.

The differences in the results may be attributed to different causes like, for instance, different paradigms employed for solving neutron transport (deterministic vs. stochastic), different levels of detail achieved in the model geometry (large hexagonal-z prisms vs. complex structures composed of small objects) and in cross-section data (processed few-group data vs. quasi-raw

One ALEPH composition-evolution calculation with 25 zones to burn takes approximately 100 minutes on the same machine (Malambu and Popescu, 2014).

[14] In the DIF3D calculations, the thermal energy is produced via both neutron fission and radiative capture reactions. The energy produced in the non-fissile region accounts approximately for 2 % of the total energy produced.

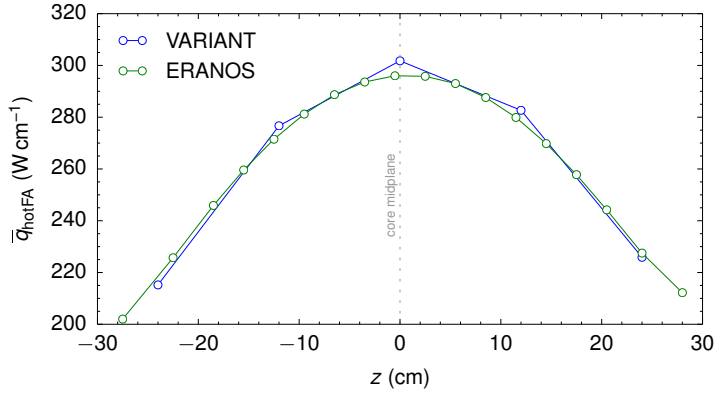


Figure 3.7: Linear power axial profile of the average pin in the hot FA (Case 1): results 1; VARIANT and ERANOS (*source:* Sarotto, 2011) results

point-wise data), or different base nuclear data libraries[15].

In general, it can be concluded that the core characteristics calculated using the VARIANT code are in a good agreement with the results obtained using more sophisticated tools, especially considering the simplicity of the adopted neutronics model described earlier in this chapter.

The biggest advantage of the presented modeling approach are the low computation time requirements; one calculation takes approximately 7.4 s on a Dell Latitude E6510 laptop with an Intel® Core™ i5 M520 processor (4×2.4 GHz) and 3.7 GiB of RAM. This allows for solving the MYRRHA loading pattern optimization problem by evaluating thousands of core configurations as described in Section 2.2.4, within a reasonable time-frame. For example, 10 000 evaluations would take less than one day, if executed as a serial job on a single CPU. However, this is true only for simple BOC problems with one criticality calculation. Should depletion analysis be involved, at least two criticality calculations are needed, doubling the computation time or more. This would lead to unacceptable problem solving times without a parallelized optimization tool, not mentioning the fact that a much higher number of trial evaluations may be needed in certain cases. Therefore, further improvements in the reactor calculation time are desirable, for instance, by employing a diffusion solver instead of the transport one.

[15]The following nuclear data libraries were used in the calculations: ENDF/B-V, Kinsey, 1983 (DIF3D), JEFF3.1, Santamarina et al., 2009 (ERANOS), ENDF/B-VII.1, Chadwick et al., 2011 (MCNPX).

Table 3.7: MYRRHA-FASTEF BOL core characteristics (Case 1): results 2

	DIF3D10.0 solver			
Parameter	VARIANT	DIF3D-diffusion		
Nodal approximation	240/-11/0	default[a]	default[a]	default[a]
Transport approximation	no	yes	yes	yes
Boundary condition	zero $\psi^-_{\partial V}$	$\beta = 0.032$	$\beta = 0.852$	$\beta = 0.786$
N_g	15	15	15	33
R	10	10	0	0
Characteristic				
k_eff[a]	1.04346[b]	0.98773	0.98801	0.98503
q_max ($\mathrm{W\,cm^{-1}}$)	318.2	309.0	309.2	310.3
Power peaking				
Hot FA $P_{\mathrm{r\,FA}}$[b]	1.030	1.030	1.030	1.030
Hot FA P_z	1.169	1.157	1.157	1.155
$P_{\mathrm{r\,core}}$	1.415	1.388	1.388	1.396
P_q	1.703	1.654	1.655	1.661
Peak ϕ ($\mathrm{n\,cm^{-2}\,s^{-1}}$)				
Central IPS	3.77×10^{15}	3.66×10^{15}	3.67×10^{15}	3.68×10^{15}
Satellite IPSs	3.26×10^{15}	3.18×10^{15}	3.20×10^{15}	3.20×10^{15}
T_clad (°C)	537.8	533.8	534.0	535.8
Computation t (s)[c]	2.9	1.4	0.3	0.8

[a] See Section 3.2.2.
[b] The value 1.030 set as a conservative estimate on the input to the calculation.
[c] Executed on a laptop with an Intel® Core™ i5 M520 processor (4×2.4 GHz) and 3.7 GiB of RAM. Approximate values.

DIF3D-diffusion calculations. The diffusion theory may be considered as a valid approximation for fast-reactor analysis in some cases, as pointed out in Section 2.3.1. For instance, the diffusion nodal option may be used in the parametric physics design studies as stated by W. S. Yang and Khalil (2001). Moreover, neutron diffusion physics codes have been routinely used for real industrial fast-reactor fuel management applications in the past (e.g., Ohkawachi et al., 2003). In this part, the DIF3D nodal diffusion solver (DIF3D-diffusion) was applied for solving the MYRRHA-FASTEF BOL criticality problem.

It can be seen from Table 3.7 that the results obtained using the DIF3D-diffusion solver are very similar to those obtained from the reference VARIANT calculation (cf. Table 3.6), while reducing the computation time by a factor of ≈ 5. The application of the albedo outer-boundary condition (2.37) makes it possible

to reduce the computation time further by simulating the back-reflection of the neutrons leaving the core region on the surrounding coolant, and thus, decreasing the size of the model. This can be achieved by setting up adequate albedo values β_g from Eq. (2.38). Two sets of results for 15 and 33 energy groups are shown in Table 3.7 for a modified MYRRHA-FASTEF model without any LBE outside of the core barrel ($R = 0$). For the sake of simplicity, the same value of albedo was used for all energy groups in both cases, that is, $\beta_g = \beta$ for $g = 1, 2, \ldots, N_\mathrm{g}$. These values are indicated in Table 3.7 and were obtained by minimizing the sum

$$\sum_i \left| \frac{\phi_i - \phi_{\mathrm{ref}\,i}}{\phi_{\mathrm{ref}\,i}} \right| , \tag{3.2}$$

where ϕ_i is the average neutron flux in a cell i calculated for a given albedo value, $\phi_{\mathrm{ref}\,i}$ is the average neutron flux in the same cell coming from the VARIANT calculation from Table 3.6, and i goes through all mesh cells in the fissile and IPS irradiation region. The maximum difference between two corresponding values was lower than 1.9 % and the absolute difference per a mesh cell was lower than 0.7 % on average in both cases. To illustrate the impact of the transport correction on the results, a VARIANT calculation was performed with total cross sections instead of the transport ones and the results were included in Table 3.7.

Figures 3.8 and 3.9 show the neutron flux distribution in the central core section and the average linear power distribution in the hot fuel assembly for all four calculations. As expected, all results are nearly identical except from the VARIANT ones.

3.3.2 Case 2

Both BOC and EOC characteristics of the BOL MYRRHA fuel loading are given as the loading pattern consists of fresh fuel assemblies only. All subsequent loadings though will consist of fuel assemblies of different burnups (or ages) and thus will give the reactor core different BOC and EOC neutronics qualities depending on the way they will be organized in the core, that is, on the loading pattern. Unlike Case 1, Case 2 focuses on the neutronics analysis of a MYRRHA-FASTEF core loading composed of fuel assemblies of different ages; in particular, the reference critical equilibrium loading made of 69 fuel assemblies organized in 5 fuel batches and described in Section 2.4.4 was analyzed. As a reminder, the corresponding loading pattern is shown in Figure 3.10 for both BOC and EOC stages.

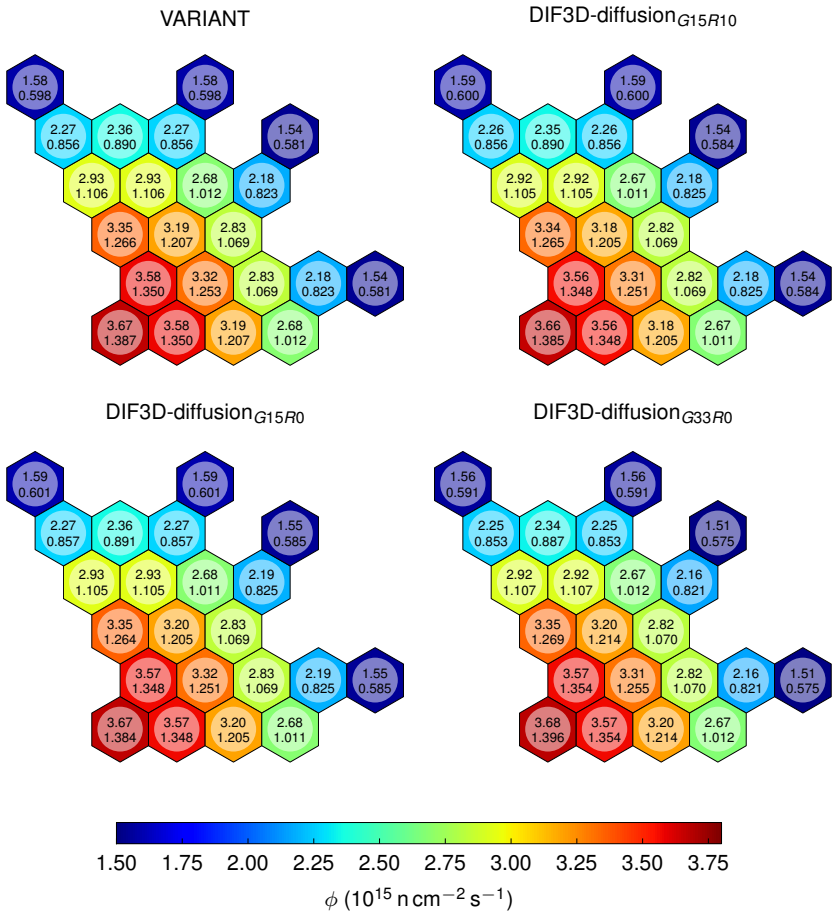


Figure 3.8: ϕ map (top values) and form factors (bottom values) for FA and IPS channels (Case 1): results 2; VARIANT results ($N_g = 15$, $R = 10$); DIF3D-diffusion results for $N_g = 15$, $R = 10$, and $\beta = 0.032$ ($G15R10$), $N_g = 15$, $R = 0$, and $\beta = 0.852$ ($G15R0$), and $N_g = 33$, $R = 0$, and $\beta = 0.786$ ($G33R0$); all values for a 12 cm-thick central core region

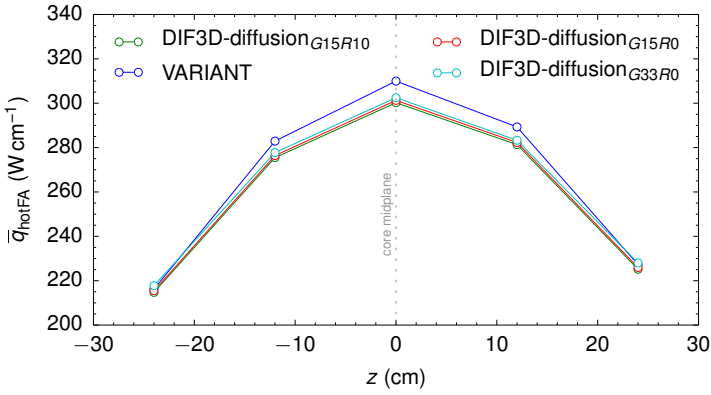


Figure 3.9: Linear power axial profile of the average pin in the hot FA (Case 1): results 2; VARIANT results ($N_{\text{g}} = 15$, $R = 10$); DIF3D-diffusion results for $N_{\text{g}} = 15$, $R = 10$, and $\beta = 0.032$ ($G15R10$), $N_{\text{g}} = 15$, $R = 0$, and $\beta = 0.852$ ($G15R0$), and $N_{\text{g}} = 33$, $R = 0$, and $\beta = 0.786$ ($G33R0$); all values for a 12 cm-thick central core region

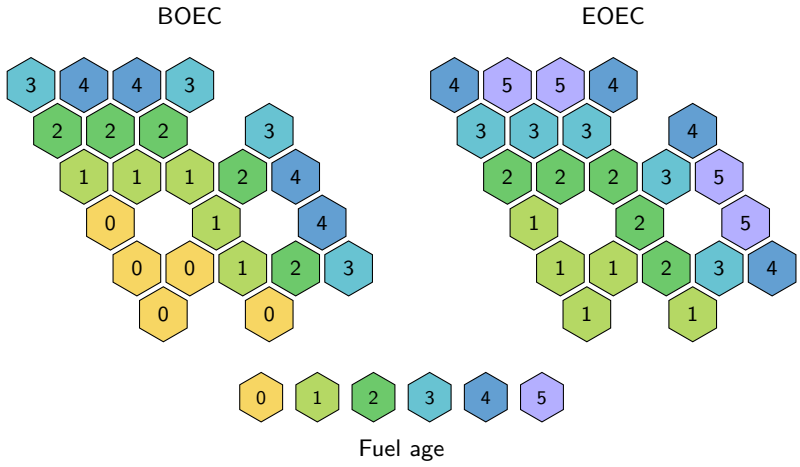


Figure 3.10: MYRRHA-FASTEF BOEC and EOEC loading patterns

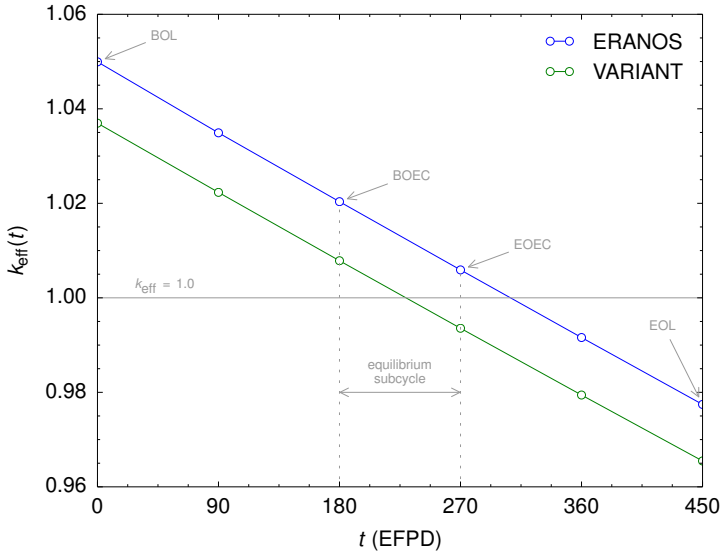


Figure 3.11: 1-batch-1-cycle burnup evaluation (k_{eff}); VARIANT and ERANOS (*source:* Sarotto, 2011) results

1-Batch-1-Cycle Burnup Evaluation

In order to determine fuel compositions corresponding to five different fuel batches, the neutronics model exploited a simplifying hypothesis concerning the equilibrium core depletion scheme. This hypothesis was adopted from Sarotto, 2011 and it relies on so-called "1-batch-1-cycle burnup evaluation" carried out on the 100 MW MYRRHA-FASTEF core composed of 69 fresh fuel assemblies. In this way, the approximate BOEC fuel-batch compositions are obtained as the core-averaged fuel compositions at the beginning of five consecutive 90 EFPD-long burnup steps during the uninterrupted whole-core depletion evaluation[16].

The results of the 1-batch-1-cycle burnup evaluation obtained by ERANOS and REBUS[17] codes are shown in Figure 3.11 for the k_{eff} and in Table 3.8 for the fuel composition evolution. It can be seen from Figure 3.11 that the slopes of the curves are nearly identical for both codes and that the REBUS calculation gives k_{eff} values consistently underestimated by ≈ 1250 pcm comparing to the

[16] All CRs were fully withdrawn in the calculation.

[17] The 240/-33/0 VARIANT solver options were used in the calculation with 33 energy groups.

Table 3.8: 1-batch-1-cycle burnup evaluation (fuel composition in kg)

Code	Elem.	Time (EFPD) 0	90	180	270	360	450
REBUS	U	735.72	731.58	727.38	723.13	718.81	714.44
	Pu	381.11	375.61	370.21	364.91	359.69	354.56
	Np	0.00	0.02	0.03	0.05	0.07	0.09
	Am	6.39	6.69	6.96	7.22	7.47	7.71
	Cm	0.00	0.10	0.19	0.26	0.33	0.39
	MAs	6.39	6.80	7.18	7.54	7.87	8.18
	FPs	0.00	9.25	18.50	27.73	36.95	46.17
ERANOS[a]	U	735.57	731.47	727.30	723.08	718.79	714.45
	Pu	381.05	375.15	369.49	363.94	358.47	353.10
	Am	6.39	6.69	6.97	7.23	7.48	7.71
	MAs	6.39	6.96	7.36	7.73	8.08	8.41
	FPs	0.00	9.20	18.38	27.55	36.72	45.87

[a] *Source:* Sarotto, 2011.
MAs = Np + Am + Cm.

ERANOS ones[18]. Also, the fuel compositions are very similar for both codes as follows from Table 3.8. Both the similar slopes of the curves from Figure 3.11 (only a ≈ 100 pcm difference over 450 EFPD) as well as the similar values from Table 3.8 support the validity of the simplified depletion scheme and fission products treatment in the REBUS depletion model described earlier in Section 3.2.3.

Core Characteristics

In the following, the neutronics characteristics of the reference MYRRHA-FASTEF equilibrium critical core obtained using the models and tools from Section 3.2 are presented and compared with those obtained using other tools, in particular the ERANOS and MCNPX/ALEPH codes.

Two different approaches were employed to get the fuel composition of different fuel batches. The first one was based on the 1-batch-1-cycle burnup evaluation and used in the REBUS and ERANOS calculations. The other one used in the MCNPX/ALEPH calculations was based on the explicit equilibrium reshuffling

[18] Most of this difference is attributed to different nuclear data libraries used by the codes.

Table 3.9: MYRRHA-FASTEF BOEC core characteristics (Case 2)

	Neutronics code			
Parameter	VARIANT	DIF3D-diff.	ERANOS	MCNPX
Nodal approximation	240/-33/1	default[a]		-
Boundary condition	zero $\psi^{-}_{\partial V}$	$\beta = 0.852$		
R	10	0		
CRs position	out	out	14 cm in	out
Characteristic				
k_{eff}[b]	1.02931	1.01636	1.01501	1.01580 $\pm$0.00011
$q_{\max}$ (W cm^{-1})	322.2	320.9	329.5	340.7
Power peaking				
Hot FA $P_{\text{r FA}}$	1.030[c]	1.030[c]	1.021	1.024
Hot FA P_{z}	1.156	1.155	1.153	1.152
$P_{\text{r core}}$	1.446	1.441	1.462	1.507
P_{q}	1.722	1.715	1.721	1.777
Peak ϕ (n cm^{-2} s^{-1})				
Central IPS	3.79×10^{15}	3.81×10^{15}	3.88×10^{15}	3.79×10^{15}
Satellite IPSs	3.36×10^{15}	3.38×10^{15}	3.44×10^{15}	3.31×10^{15}
T_{clad} (°C)	558.3	557.5		
Computation t (s)[d]	9.7	0.4		–[e]

[a] See Section 3.2.2.
[b] The MCNPX result is in the $\mu \pm \sigma$ format.
[c] Set as a conservative estimate on the input to the calculation.
[d] Executed on a laptop with an Intel® Core™ i5 M520 processor (4×2.4 GHz) and 3.7 GiB of RAM. Approximate values.
[e] See Footnote 13 on page 83.
Source: Sarotto, 2011 (ERANOS); Sarotto, Fernandez, et al., 2012 (MCNPX).

scheme repetitively applied to the core with 69 fresh fuel assemblies until the core characteristics converged (Sarotto, Fernandez, et al., 2012).

BOEC results. The results of the BOEC analysis are summarized in Table 3.9. The control rods were fully withdrawn from the core in all cases except from the ERANOS calculation, in which the control rods were partially inserted in the fissile zone (14 cm) to compensate for the ≈ 1500 pcm k_{eff} swing over the equilibrium cycle. The VARIANT and DIF3D-diffusion calculations were performed with 15-group nuclear data.

As can be noticed from Table 3.9, the DIF3D-diffusion results are closest to the reference MCNPX ones in terms of k_{eff} estimations. The ERANOS value for the "all-rods-out" condition, $k_{\text{eff}} = 1.03072$ (Sarotto, 2011), is very similar to the VARIANT value[19]. The values of all other characteristics from the table also stay reasonably close to the reference ones for all the codes, the relative difference being typically lower than 6 %. The similarity between the neutron flux distributions obtained using different codes is demonstrated in Figure 3.12. Again, significantly less time was needed for one DIF3D-diffusion calculation than for one VARIANT calculation (almost 25-times less in this case).

EOEC results. The end-of-equilibrium-cycle (EOEC) results obtained for the BOEC core depleted for 90 EFPD are summarized in Table 3.10. Obviously, the control rods were withdrawn from the core in all cases. The REBUS values denote the results obtained using the DIF3D-diffusion solver for criticality calculations inside the REBUS fuel cycle code. The REBUS k_{eff} estimation is very similar to the reference MCNPX value (147 pcm overestimation). Comparing the REBUS and ERANOS power-peaking factors, it can be noticed that the power density distribution is less flat for REBUS than for ERANOS, leading to slightly larger peak neutron fluxes in IPS channels in case of the first of the codes. The computation time needed for one REBUS calculation is lower than two seconds, which makes it well suitable for MYRRHA-FASTEF loading pattern optimization.

Figure 3.13 shows the neutron flux distribution in the central core region calculated using the REBUS and ERANOS codes. The differences between the two sets of values are caused mainly due to different methodologies used to obtain them (region-averaged vs. maximum values). Figure 3.14 compares the $\bar{q}_{\text{hotFA}}$ profile for different codes and both BOEC and EOEC states. It can be seen that all curves are very similar, the BOEC curves being positioned slightly higher than their EOEC counterparts. The only exception is the ERANOS BOEC curve, with the peak shifted in the upward direction and larger values in the upper core part. These two effects can be explained by the partial insertion of the control rods into the fuel zone (from bottom), not considered in the other models.

3.4 Summary

The third chapter presents reactor analysis models and computing tools to be used for MYRRHA-FASTEF loading pattern optimization. In particular,

[19] Both codes use the same nodal variational transport neutronics solver.

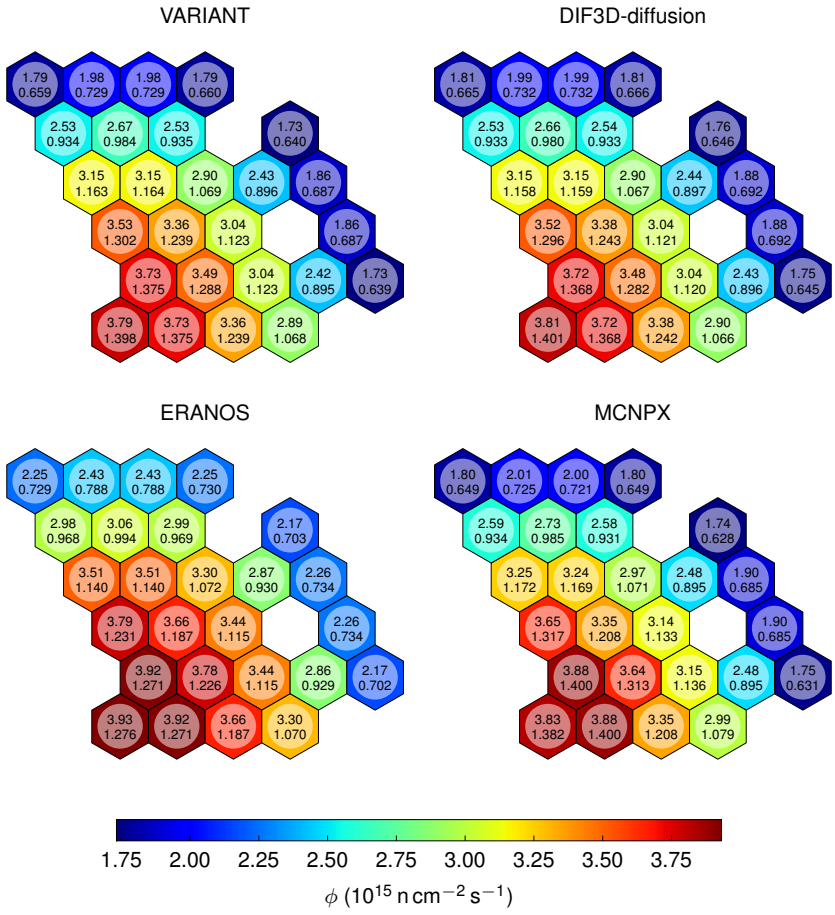


Figure 3.12: BOEC ϕ map (top values) and form factors (bottom values) for FA and IPS channels (Case 2); VARIANT results for a 12 cm-thick central core region (CRs out, $R = 10$); DIF3D-diffusion results for a 12 cm-thick central core region (CRs out, $R = 0$, $\beta = 0.852$), ERANOS results (CRs 14 cm in): maximum values (*source:* Sarotto, 2011); MCNPX results (CRs out): mean values obtained for 12 cm-high tally volumes inside the stiffness pins (*source:* Sarotto, Fernandez, et al., 2012)

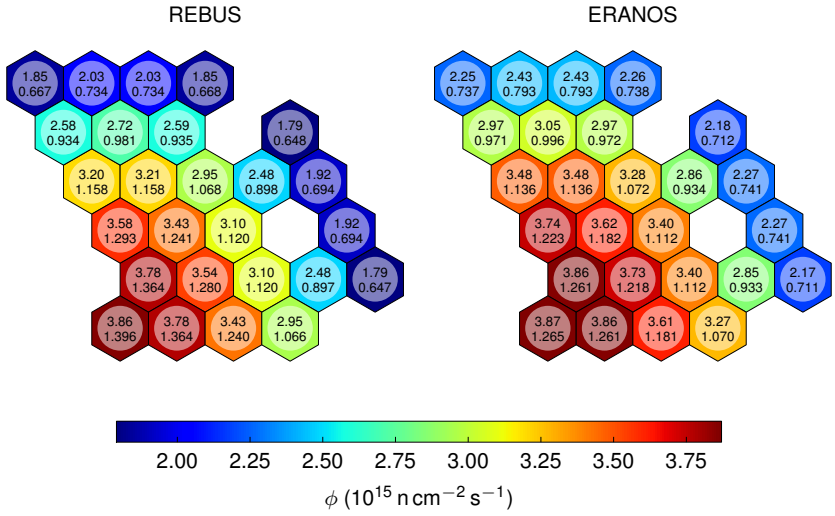


Figure 3.13: EOEC ϕ map (top values) and form factors (bottom values) for FA and IPS channels (Case 2); REBUS results for a 12 cm-thick central core region; ERANOS results: maximum values (*source:* Sarotto, 2011)

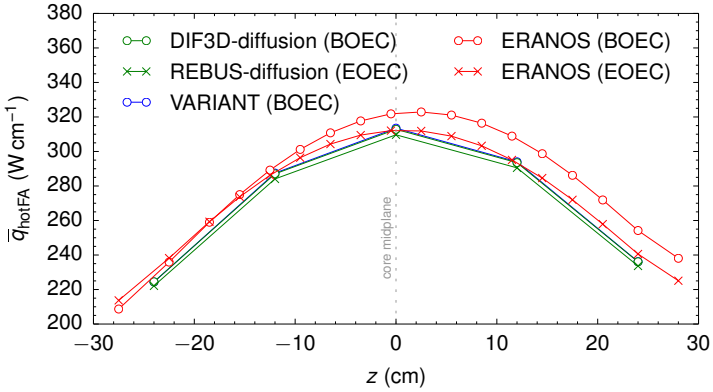


Figure 3.14: Linear power axial profile of the average pin in the hot FA (Case 2); VARIANT BOEC results (CRs out, $R = 10$); DIF3D-diffusion BOEC and REBUS EOEC results (CRs out, $R = 0$, $\beta = 0.852$), ERANOS BOEC (CRs 14 cm in) and EOEC (CRs out) results (*source:* Sarotto, 2011)

Table 3.10: MYRRHA-FASTEF EOEC core characteristics (Case 2)

	Neutronics code		
Characteristic	REBUS	ERANOS	MCNPX
k_{eff}[a]	1.00181	1.01612	1.00034 ±0.00004
$q_{\max}$ (W cm^{-1})	317.8	329.5	
Power peaking			
Hot FA $P_{\text{r FA}}$	1.030[b]	1.020	
Hot FA P_{z}	1.155	1.144	
$P_{\text{r core}}$	1.427	1.423	
P_{q}	1.698	1.662	
Peak ϕ (n cm^{-2} s^{-1})			
Central IPS	3.86×10^{15}	3.82×10^{15}	
Satellite IPSs	3.43×10^{15}	3.41×10^{15}	
T_{clad} (°C)	554.6		
Computation t (s)[c]	1.7[d]		–[e]

[a] The MCNPX result is in the $\mu \pm \sigma$ format.
[b] Set as a conservative estimate on the input to the calculation.
[c] Executed on a laptop with an Intel® Core™ i5 M520 processor (4×2.4 GHz) and 3.7 GiB of RAM. Approximate values.
[d] Comprises two criticality calculations, BOC and EOC.
[e] See Footnote 13 on page 83.
Source: Sarotto, 2011 (ERANOS); Sarotto, Fernandez, et al., 2012 (MCNPX).

the MC^2-2, DIF3D10.0, and REBUS-PC 1.4 fast-reactor analysis codes for few-group cross sections generation, whole-core calculations, and depletion analysis, respectively, are presented together with the detailed description of tailored MYRRHA-FASTEF models for calculation of the key neutronics characteristics: k_{eff} and neutron flux and power density distributions.

It is shown that the opted modeling approach and elaborated models give results yet accurate enough for MYRRHA-FASTEF loading pattern optimization purposes, while requiring reasonably short computation times. A single BOC criticality calculation of the critical equilibrium MYRRHA-FASTEF core layout using the variational nodal transport VARIANT solver takes less than 10 seconds on a reference computer and less than 0.5 seconds using the DIF3D nodal diffusion solver and tuned outer-boundary albedo condition (see Table 3.9). The

employment of the diffusion theory becomes especially beneficial when solving depletion problems that need multiple criticality calculations. The proposed REBUS neutronics model takes less than two seconds for the MYRRHA-FASTEF EOC core evaluation (see Table 3.10).

A special module for thermal-hydraulics analysis of MYRRHA-FASTEF loading patterns is also briefly presented in Appendix B. This module accepts power peaking factors, total number of fuel assemblies, and reactor thermal power as its input variables and returns estimated maximum cladding temperature on its output—the most important parameter for the preliminary thermal-hydraulics assessment of the evaluated core configuration.

Chapter 4

MYRRHA Loading Pattern Optimization

In this chapter, the MYRRHA loading pattern optimization problems are presented. The optimization goals are formulated and the opted solution approach is explained and justified first, based on the extensive information given in Chapter 2. This is followed by a description of the applied optimization methods. The optimization problems are then solved in the last part of the chapter.

The optimization problems discussed hereinafter concern only the critical MYRRHA core design. The same problems with adapted constraints (e.g., without the EOC criticality condition) could be solved also for the subcritical core, however, this would require a use of special tailored neutronics models, not described in Section 3.2.

4.1 Optimization Goals

A thorough general discussion of a variety of in-core fuel management optimization objectives, decision variables, and constraints for different types of nuclear reactor installations was presented in Section 2.1.2. More specific information with regard to the MYRRHA in-core fuel management was then given in Section 2.4.5 on page 57. Based on this information and considering the primary purpose of the facility, which is to provide fast and thermal neutrons for science and engineering applications, the MYRRHA loading-pattern

optimization objective was defined as maximizing the neutron fluence achieved in the experimental rigs in the IPS channels during a standard single fuel cycle of 90 EFPD.

Three types of constraints are to be considered in the problem: the reactivity limit saying that the reactor core must be critical at the EOC, the maximum cladding temperature limit as the most severe thermal limit to ensure the integrity of the fuel during operation (see the paragraph on the working temperatures in Section 2.4.4), and the number of fuel assemblies of different ages to be inserted into the core before the beginning of the optimized fuel cycle. It is presumed that the MYRRHA core will be completely unloaded after each fuel cycle and that none of the fuel assemblies will be required to stay at its previous position. This way, all fuel assemblies are allowed to move freely within the core, when designing a new loading pattern. The azimuthal orientation of the fuel assemblies is neglected. It is also assumed that the shutdown margins are met for every loading pattern, using the same shutdown system as the one described in Section 2.4.4 on page 52. Therefore, it is not necessary to evaluate loading patterns for different "rodded" conditions as shown in Table 2.5.

The MYRRHA-FASTEF loading pattern optimization problem is described in detail as Problem 2 below in Section 4.4.2.

4.2 Solution Approach

The optimization methods for tackling the loading pattern optimization problem were divided into two distinct groups in Section 2.2.4: methods relying on a mathematical model of the problem and iterative methods employing standalone reactor analysis codes. Considering the special character of MYRRHA, its flexible design, and the wide spectrum of its applications, it has been decided since the very beginning of the PhD project to utilize methods of the second group for MYRRHA core-reload optimization. As explained in Section 2.2.4, making use of reactor analysis codes for core-reload optimization allows to avoid the principal problems associated with rigorous mathematical programming methods (i.e., excessive simplifications in the fuel cycle and reactor physics models and the need for an expensive preprocessing step), while providing desired freedom with respect to varying reactor-core design (critical vs. subcritical design, variable number of FAs, etc.) and optimization objectives and constraints. The general optimization scheme characteristic for this optimization approach can be represented by the two-block system depicted in Figure 4.1; one block is a "black-box" optimization algorithm that modifies the current-generation loading pattern(s) to improve it (them) based on the quality of its (their) predecessor(s),

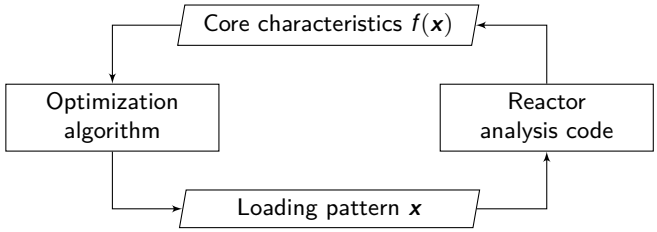


Figure 4.1: General iterative loading-pattern optimization scheme

and the second one is a black-box reactor analysis code that evaluates different core-loading design(s) coming from the output of the optimization algorithm and then support it back with the calculated characteristics on its input.

The subgroup of metaheuristic optimization methods have been considered primarily for the application in this work, as they represent the most efficient state-of-the-art methods applied to the loading pattern optimization problem.

4.3 Optimization Methods

Two metaheuristic methods were selected and implemented for solving the MYRRHA-FASTEF loading pattern optimization problem mainly to demonstrate the capability of the RELOAD-M optimization tool described in the following chapter to accommodate different optimization methods. The first one is Genetic Algorithm (GA), an example of a well-established evolutionary metaheuristic optimization method that has been widely used for nuclear-reactor fuel-loading optimization purposes, including applications in several production codes (see Table A.1). The second method is Ant Colony Optimization (ACO), a more recent representative of metaheuristics applied to the reshuffling problem, essentially different in its nature from GA. Both methods fall within the group of nature-inspired population-based metaheuristics (see the paragraph on metaheuristics in Section 2.2.4, page 30).

The GA and ACO used further in the chapter for solving the MYRRHA-FASTEF loading pattern optimization problems are described in detail in the following sections, adopting the terminology used in Talbi, 2009. Note that the presented algorithms are rather basic and do not exploit advanced concepts such as hybridization, parallelism, intelligent parameter-tuning techniques, dynamic variations on the optimization parameters, or the Pareto optimality concept for tackling multi-objective, respectively, constrained problems. Application of

theses techniques in the current work would go beyond its scope, and hence, should be considered rather as a standalone theme for a separate follow-up research project.

4.3.1 Genetic Algorithm

GA belong to the EA family of population-based metaheuristics. The EAs are inspired by the principles of natural evolution first presented by Charles Robert Darwin in his famous book On the Origin of Species (1859). According to Darwin's revolutionary theory of biological evolution, all species of organisms arise and develop through the natural selection of small, inherited variations that increase the individual's ability to compete, survive, and reproduce. This theory was later supported by Johann Gregor Mendel's discoveries of the laws of inheritance, the foundation of modern genetics, which provided the needed scientific basis and instrumental framework. The Darwinian principles of variation, selection, and inheritance (or replication, retention) were translated in the 1980s by computer scientists into three algorithmic procedures essential for all EAs: selection, recombination & mutation reproduction, and population replacement. Depending on a particular construction of these procedures (and specifics of the solved problems and other features), one talks about different EAs such as GAs, ESs, EPs, or GPs (e.g., Talbi, 2009). Among them, GAs are arguably the best-known and most-studied discrete optimization methods, which, ergo, makes them also the best-suited EAs for this work.

GAs possess the following characteristics: it is a "weak" (i.e., applicable to many different problems) but powerful method that works well with high combinatorial dimensionality, nonlinear discrete objectives and constraints, multimodality, and lack of direct derivative information (Poon and Parks, 1993). It is also regarded as a robust method, that is, a method relatively insensitive to the optimized problem (DeChaine and Feltus, 1995a). Another intrinsic feature is the ability to simultaneously search different regions in the search space, giving it an advantage over global optimization compared to other methods. Good results are usually achieved with a minimum human problem-knowledge embedded in the algorithm, which is a typical characteristic of black-box methods. Being a typical population-based technique, GA is also well-suited for implementation on parallel computers. Among its disadvantages one should mention the generally-know fact that GA performs poorly for linear problems and that the optimum solution is not guaranteed to be found. Also, it is generally accepted that GAs are better at fitting the neighborhood of the optimum solution than at locating the exact optimum (Holland, 1975), as will be pointed out later.

Many GA variants have been used to solve the loading pattern optimization

problem for different types of reactors (see Table A.1), including PWRs (e.g., Alim, Yilmaz, et al., 2009; Boroushaki et al., 2003; Chapot et al., 1999; Dai et al., 2012; Guler et al., 2004; Karahroudi et al., 2013; Norouzi et al., 2013; Parks, 1995, 1996, 1997; Poon and Parks, 1992; Yamamoto, Noda, et al., 1997), BWRs (e.g., J.-L. François and López, 1999; Kobayashi and Aiyoshi, 2002, 2003b; Martín del Campo, J. L. François, et al., 2004; Ortiz and Requena, 2004a), PHWRs (Do et al., 2006a,b), GCRs (Ziver, Pain, et al., 2004), and fast reactors (Toshinsky et al., 1999). It has been stated by some authors that GAs perform generally well in solving the core reload problem, compared to other metaheuristics such as the single solution-based SA[1]. For instance, Carter (1997), DeChaine and Feltus (1995a), Poon and Parks (1993), and Yamamoto (1997) agreed that GA makes a quicker initial progress at global search, but loses its performance later during the optimization, when SA may become more efficient, improving the objective function more consistently at local search. Thus, GA can deliver a high-quality solution faster than SA, when a relatively short computation time is the limiting factor. However, these conclusions are very premature and to confirm or reject them, a more conclusive benchmark study involving a broader spectrum of methods would be required that has not been done yet[2]. Carter (1997) suggested GA as the most suitable method among other metaheuristics (DS, GDA, TS, SA, PBIL) to perform regular loading-pattern optimization studies. For final optimization, he recommended to couple the algorithm with a simple hill-climbing method to improve its local-search capability.

Before all parts of the GA will be described closer, it is useful first to explain the special jargon commonly used within the EA community and interpret it in the context of the nuclear-reactor loading-pattern optimization. Another important aspect of all metaheuristics that needs a particular attention is the encoding of a solution, described afterward. A reader interested in general aspects of the application of the GA and other metaheuristics to the loading pattern optimization problem is referred to Carter, 1997; Meneses, Lima, et al., 2010.

[1] Comparisons with other metaheuristics in solving the LPOP are very rare in literature. Khoshahval, H. Minuchehr, et al. (2011) compared GA with PSO in their paper and concluded that both techniques are comparable, GA being faster than PSO, while requiring more computation time. Similarly, Hill and Parks (2015) compared GA with TS and SA. In this case it is stated that TS outperforms both other algorithms.

[2] The summarizing work of J.-L. François, Ortiz-Servin, et al. (2013) may be viewed as the first attempt at such a study. It compares performances of seven metaheuristics in optimizing an equilibrium BWR core-reload pattern. These are the GA, TS, Recurrent Neural Network (RNN), and four ant-colony algorithms: Ant System algorithm (AS), Max-Min Ant System (MMAS), Best-Worst Ant System (BWAS), and Ant Colony System (ACS). (See Table A.1 for the corresponding literature references.) GA was outperformed by TS and AS in this particular study.

Terminology

A lot of the EA terminology has been adopted from genetics, a field of biology that studies genes, heredity, and variation in living organisms (Griffiths et al., 2010). The fundamental terms taken from genetics and used in the EAs, often in a customarily inadequate or simplified way, comprise the gene, a discrete heritable unit determinant of the inherent properties of species passed from parents to progeny; genome, an organism's total complement of genes; chromosome, a complex structure containing an organized set of genes; locus, the specific location of a gene on a chromosome, which determines what organism's trait can be influenced; allele, a variant of the gene at a given locus, which determines how the organism's trait can be influenced; genotype, the allelic constitution of an organism; and phenotype, the set of the organism's observable traits. Having a nuclear-reactor loading pattern as the subject of the GA optimization, the gene is loosely interpreted as a fuel assembly or a fuel cell in the core, making the set of all available fuel assemblies (fuel cells) the genome; the chromosome is perceived as the representation of a loading pattern; the locus is an equivalent of the position of the fuel assembly in the loading pattern; the allele is seen as the type (value) of the fuel assembly; the genotype corresponds to the set of all fuel assemblies used in the loading pattern; and the phenotype is an analogue of all loading-pattern characteristics (traits), such as the fissile-material content, k_{eff}, power-peaking factors, neutron flux and power density distributions, etc.

Other frequently used terms are the individual, a single organism with a unique phenotype; population, a group of individuals; generation, a population of individuals living during the same time; parents, a group of individuals of one generation from which another group of individuals called children (or progeny, offspring) arises as the result of the gene combination, constituting a new generation. In the GA terminology, the individual corresponds to a candidate or a trial solution represented by a loading pattern, in this particular case, and the population corresponds to a set of chromosomes updated at each iteration, that is, at each generation.

Of the utmost importance in the evolutionary theory is the notion of fitness, which is defined as an individual's ability to propagate its genes in the future generations. In the EAs, fitness is the value of the objective (fitness) function used as a single figure of merit to express the quality of a solution in terms of achieving the optimization goals (cf. Eq. (2.1) on page 22).

Representation

The problem representation (coding) is the single most critical issue in the implementation of a GA and together with the process of converting a solution into a chromosome, called encoding, and the reverse process, called decoding, it plays a major role in the efficiency and effectiveness of any metaheuristics (Talbi, 2009). The general rule when picking the right representation is that it should be as natural and simple as possible and preserve important relationships between solution elements (Carter, 1997; Poon and Parks, 1993).

Many representations have been proposed over the time for different optimization problems. The binary string coding was originally used by Holland (1975) in the canonical version of his GA, which maps each gene of a solution to a string of a given length, filled with zeros and ones only (binary bits). However, this representation is not natural for ordering and permutation problems such as the well-known Traveling Salesman Problem (TSP) (Applegate et al., 2007), a classical optimization problem conceptually similar to the loading pattern optimization problem, and hence, may fail to give good results[3,4]. Therefore, special ordered-list representations have been designed for "TSP-class" problems like the permutation representation, adjacency-listing representation, position-listing representation, and others (e.g., Poon and Carter, 1995).

Probably the best-fitted representation for the loading pattern optimization problem is that of a two-dimensional grid, because it preserves all the relationships most naturally. The representation most frequently found in literature, however, projects a loading pattern onto a discrete linear sequence of integers—the integer-string representation. Comparing to the permutation coding[5], another well-suited loading-pattern representation, the integer-string representation can hold more information, as it can represent more solutions ($N! < N^N$). In both cases, the alleles are expressed in the form of unique integer numbers, one integer per one fuel-assembly type. Less common, but also possible,

[3]The binary string representation has been used by some authors for solving the LPOP (e.g., DeChaine and Feltus, 1995a; Hongchun, 2001). The main drawback of this approach lies in the bias introduced into the solution genotype, when the number of the fuel types N is not a power of two, $N \neq 2^n$ for $n \in \mathbf{N}_0$. Although this bias (some FA types being associated with more alleles than others) cannot be eliminated for binary genotypes, DeChaine and Feltus (1996) reduced it by adding additional bits to the genes.

[4]The TSP is usually defined as follows: Given a list of towns and the distances between each pair of them, find the shortest possible route starting from a given town, passing through all the other towns exactly once, and returning to the origin town.

The analogy with the LPOP is straightforward; instead of the towns to visit, the problem is the arrangement of FAs and a complete tour is equivalent to a pattern of assemblies in the reactor core.

[5]Unlike in the TSP, partial permutations $P(n, N)$ can be used in the LPOP, if the number of available FA types N exceeds the number of fuel batches n, $N > n$.

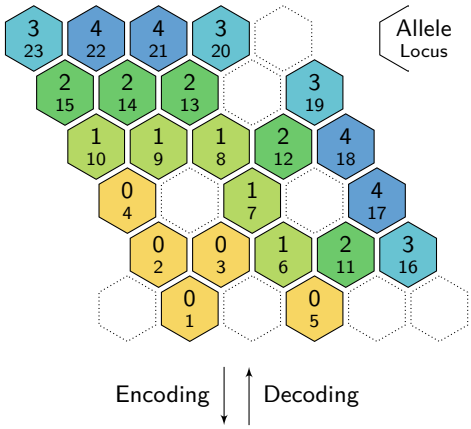
are representations that manipulate with real numbers. For instance, Meneses, M. D. Machado, et al. (2009) used the Random Keys technique proposed by Bean (1994) to represent different loading patterns. This approach allows to avoid potential difficulties associated with the application of the crossover operator, which will be discussed later.

Obviously, the number of possible problem representations is unlimited and its choice/design depends merely on the programmer, who, besides the representation's naturalness to the problem, also looks at the design of a suitable crossover operator and the amount of the problem-related information it should hold. An example of an advanced loading-pattern representation, which contains also explicit information on the symmetry of each assembly position, is the integer-based array representation by Alim (2006).

In this work, the integer-string representation was used that maps fuel-assembly positions (genes) onto a vector (chromosome) of integers (alleles) in a given order. Figure 4.2 shows the mapping between the critical equilibrium MYRRHA-FASTEF core and its representation, taking advantage of the core's one-third rotational symmetry. The mapped part of the core consists of $N_{\mathrm{FA}} = 23$ unique positions filled with fuel assemblies of $N = n = 5$ types; the first, second, and third type being present five-times in the core, and the fourth and fifth type four-times, $\boldsymbol{l} = (5, 5, 5, 4, 4)$. The fuel-assembly types $\boldsymbol{t} = (t_1, t_2, \ldots, t_N)$ correspond to different fuel burnups ranked in an ascending order: $t_1 < t_2 < \ldots < t_N$. Since the 1-batch-1-cycle approach was used to obtain fuel compositions for five fuel batches present in the MYRRHA-FASTEF core, which are thence equidistantly graded with respect to their burnups, the fuel-assembly types can be interpreted as the numbers of times the fuel assemblies have resided in the core. Thus, the first type denoted by the integer 0 are the fresh fuel assemblies, type 1 are the once-irradiated fuel assemblies, type 2 are the twice-irradiated fuel assemblies, an so on: $\boldsymbol{t} = (0, 1, 2, 3, 4)$.

Algorithm

The traditional GA formally introduced by Holland (1975) is presented in Algorithm 4.1. It consists of several components—population initialization, selection strategy, reproduction strategy represented by crossover and mutation operators, replacement strategy, and stopping criterion. The algorithm proceeds in the following way: First, the initial population $\mathcal{P}$ (generation $g = 0$) of p individuals is generated; each individual $\boldsymbol{x}$ in the population is evaluated, its fitness x is assigned to it, and the population-best individual $\boldsymbol{x}_{\mathrm{opt}}$ and its fitness x_{opt} are recorded and stored in the memory at the same time. Then, s sets $\mathcal{S}$ of ν parents are selected from the population, applying the selection operator. For



$$\boldsymbol{x} = (0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 3, 4, 4, 3, 3, 4, 4, 3)$$

Figure 4.2: Loading pattern representation: the value of the n-th component of the vector $\boldsymbol{x}$ corresponds to the allele value at locus n; the white cells represent core components that stay fixed in the reactor core (cf. Figure 3.3)

each group of parents a given number o of offspring $\mathcal{O}$ is produced, applying the crossover operator with the crossover rate $p_\mathrm{c} \in (0, 1]$. Each offspring is mutated afterward with a probability $p_\mathrm{m} \in [0, 1]$ (mutation rate), applying the mutation operator. All offspring are then evaluated, being assigned with their fitness values. If appropriate, the best-so-far individual and its fitness are updated. Finally, a new population ($g = 1$) is formed according to the population replacement strategy adopted. All steps from the application of the selection operator to the population replacement are repeated until a given condition called the stopping criterion is satisfied.

In the following, the role and the details of implementation of each of the GA components will be discussed and the variants used in this chapter for solving the MYRRHA-FASTEF loading pattern optimization problem will be described.

Population initialization. The main criterion to deal with when designing an initial population of individuals is diversification. The large diversity in initial populations is crucial for the algorithm's effectiveness and its efficiency. Talbi (2009) categorizes the population initialization strategies into four categories, each characterized by its diversity, computation cost, and quality of the solutions. These are: sequential diversification, parallel diversification, heuristic

Input : $G, p, s, \nu, o, p_c, p_m$
Output: $x_{\text{opt}}, \boldsymbol{x}_{\text{opt}}$

```
1  g ⟵ 0; P ⟵ ∅; x_opt ⟵ +∞
2  for i ⟵ 1 to p do                                   // Initialization
3      x ⟵ GenerateCandidate()
4      x ⟵ Evaluate(x, g)
5      if x < x_opt then                  // Objective function minimization
6          x_opt ⟵ x                                   // E.g., k*_eff
7          x_opt ⟵ x
8      P ⟵ P ∪ x
9  while StopCriterion(G, g, p, x_opt, P) is False do   // Stopping criterion
10     g ⟵ g + 1; O ⟵ ∅
11     for i ⟵ 1 to s do
12         S ⟵ SelectionOperator(P, ν)                  // Selection
13         r ⟵ Rand (0, 1)
14         if r < p_c then
15             Õ ⟵ CrossoverOperator(o, S)              // Crossover
16             O ⟵ O ∪ Õ
17     foreach x ∈ O do
18         r ⟵ Rand (0, 1)
19         if r < p_m then x ⟵ MutationOperator(x)      // Mutation
20         x ⟵ Evaluate(x, g)
21         if x < x_opt then
22             x_opt ⟵ x
23             x_opt ⟵ x
24     P ⟵ PopulationReplacement(P, O)                  // Population replacement
```

Algorithm 4.1: Genetic Algorithm

initialization, and random generation[6]. Simply said, the sequential and parallel diversification strategies aim to sample the initial population in the range of the fitness values (objective function image sampling) and in the space of the decision variables (objective function domain sampling), respectively. The resulting population usually attains the highest diversity, the computational cost of evaluating the intial solutions is mediocre, and the quality of the solutions is poor. The last-named characteristic can be improved by using any heuristic for formation of the initial population[7]. However, while doing so, care should be

[6] And their hybrids.

[7] An example of such heuristic may be local search or any specific knowledge related to the problem at hand. For instance, if the optimum LP is expected to be organized in multiple

```
   Input  : N_FA, N, m, t
   Output: x

1  x ∈ N_0^{N_FA}
2  for i ⟵ 1 to N_FA do
3      a ⟵ 0; b ⟵ Rand(0, M)
4      for j ⟵ 1 to N do
5          a ⟵ a + m_j
6          if b ≤ a then
7              x_i ⟵ t_j                          // Allele assignment
8              Break()
9  x ⟵ RepairOperator(x)                          // Solution repair
```

Algorithm 4.2: `GenerateCandidate()` function

taken so as to avoid biasing the results. The inclusion of best-guessed solutions into the initial population, which may improve the performance of the algorithm significantly, falls also into the same category of heuristic initialization, generally characterized by high-quality solutions and low computation cost. The main drawback of this strategy is the worst diversity achieved in the initial solutions among all categories. The last category of random generation is characterized by medium diversity in the initial solutions, large computing cost for evaluating them, and their generally low quality. Building-up an initial population from randomly generated solutions is the most frequent practice in the literature on loading-pattern optimization. It is this strategy which was adopted for this work.

The implementation of the function `GenerateCandidate()` from Algorithm 4.1 is shown in Algorithm 4.2, where $\boldsymbol{m} = (m_1, m_2, \ldots, m_N)$ is the inventory (stock) of all available fuel assemblies of N different types $\boldsymbol{t}$ that are eligible to be loaded into the reactor core. The total number of the available fuel assemblies thus equals $M = \sum_{i=1}^{N} m_i$. In Algorithm 4.2, the fuel-assembly types $\boldsymbol{t}$ are first assigned to the loading-pattern cells stochastically, in proportion to their abundances m_i in the fuel inventory, and a special repair operator is then applied to the resulting solution, which checks its compliance with the fuel-inventory constraints and modifies it to make it feasible, if needed. The `RepairOperator()` function is described in detail on page 117.

An important parameter to be decided at the stage of population initialization

enrichment zones, FA's enrichment can be checked first, before it is loaded into the core in its position, if its enrichment stays within the range of expected values. The priority table technique implemented by DeChaine and Feltus (1996) works on a similar principle.

```
   Input  : x, g
   Output: x

1  if x ∈ M then y ⟵ Read(x, M)                    // Read from the memory
2  else
3  |  infile ⟵ GenerateInput(x)
4  |  outfile ⟵ ReactorCode(infile)                // Run the reactor code
5  |  y ⟵ ParseOutput(outfile)
6  |  Write(x, y, M)                               // Write to the memory
7  x ⟵ f(y, g)                                     // Evaluate fitness
```

Algorithm 4.3: `Evaluate()` function

is the size of the population p. According to Carter (1997), "The size of the population and its relationship to the performance of the GA is one of the least well documented areas." As the same author adds in his paper, some work suggests that populations should be quite large and that the initial population should be randomly chosen, meaning slower convergence but better investigation of the search space. However, some other work has suggested that it is possible to work with small breeding populations, but that it may need a non-random initial population to ensure good convergence. In order to investigate this, Ziver, Pain, et al. (2004) performed a series of tests, while solving a GCR loading pattern optimization problem. They showed clearly that with small population sizes, $p \in \{5, 10\}$, premature convergence was observed, and that for $p = 50$ considerable improvement in optimization results were obtained. Also, it is cited in the paper that medium-size populations of $p \in [20, 30]$ individuals may be sufficient for some problems. Three different population sizes were, therefore, used in this work, $p \in \{30, 50, 80\}$, to make a small contribution to this area of research.

Fitness evaluation. Whenever an individual $\boldsymbol{x}$ is created or its chromosome is changed, its fitness has to be evaluated and assigned to it, as the only measure of its quality. This is done via the `Evaluate()` function (see the lines 4 and 20 in Algorithm 4.1), which, in this particular case, prepares the input file infile for the reactor analysis code `ReactorCode()`, then launches the code, and processes the output file outfile, when the code execution is finished. Only that part of the input file is modified, which defines the layout of the fuel assemblies of different types in the reactor core. The fitness x is calculated as $x = f(\boldsymbol{y})$, where $\boldsymbol{y} \equiv \boldsymbol{y}(\boldsymbol{x})$ are the loading-pattern characteristics obtained by parsing the output file. The reactor codes and models described in Chapter 3 are used in place of the `ReactorCode()` function in Algorithm 4.3.

Because individuals may appear during the optimization that have already been evaluated, it is useful to store all evaluated individuals with their characteristics in the memory denoted by the letter $\mathcal{M}$. In case the same individual has to be evaluated later again, its characteristics can be read from the memory instead of being taken from the output of a new full reactor calculation[8]. This simple concept indicated in Algorithm 4.3 allows to speed up the optimization process significantly, since the `Evaluate()` function is by far the most time consuming part of the whole algorithm[9].

Selection strategy. As its name implies, the selection strategy mimics the process of natural selection, implementing the fundamental Darwinian principle of the "survival of the fittest"[10]. It is responsible for selecting a group $\mathcal{S}$ of ν individuals from the current population $\mathcal{P}$ to parent new individuals for the next population. In compliance with the main principle, fitter individuals are selected for mating with a higher probability than their less fit peers, which gives them an advantage in propagating their genes to the progeny. However, the worst individuals should not be discarded and they should also be given some chance to be selected, in order to ensure higher variability in the gene pool for the future generations. In this way, the selection mechanism introduces a desired bias toward better solutions usually regarded as the "selection pressure"[11].

The parents are selected according to their fitness[12] by means of various strategies (e.g., Talbi, 2009). The following listing consists of selection schemes that have been used or considered for loading-pattern optimization: different types of fitness-proportional methods like the simple roulette(-wheel) selection (DeChaine and Feltus, 1996); rank-based methods like random pairing (Boroushaki et al., 2003) or the Baker's Ranking scheme (Poon and Parks, 1993) and its extension the (μ, λ)-scheme (Carter, 1997); some trivial schemes like random-walk selection (Carter, 1997); or the most popular scheme—tournament selection (Alim, 2006; Alim, Ivanov, and S. H. Levine, 2008a; Carter, 1997; DeChaine and Feltus, 1995b; Ziver, Pain, et al., 2004). Various information can

[8]It is better to store all the LP characteristics $\boldsymbol{y}$ in the memory than the single x value, since the form of the objective function may change during the optimization (see Section 4.4.2). This is indicated in line 7 of Algorithm 4.3, where the current generation number g is included as one of the fitness function arguments.

[9]Nearly 100 % of the optimization time is spent on the fitness function evaluations, including the pre- and post-processing parts.

[10]This phrase was first used by Spencer, 1864.

[11]Poon and Parks (1993) quantify the selection pressure as the difference between the survival probability of the best individual and the worst individual.

[12]Talbi (2009) mentions two different ways fitness values can be assigned to individuals in the EAs: the proportional fitness assignment, in which the absolute fitnesses are associated with individuals; and the rank-based fitness assignment, in which relative fitnesses (e.g., ranks in the sorted population) are associated with individuals.

```
Input  : 𝒫, N_FA, ν, μ
Output: 𝒮

1  𝒮 ⟵ ∅
2  for i ⟵ 1 to ν do
3      x ∈ N_0^{N_FA}, x ⟵ +∞
4      for j ⟵ 1 to μ do                          // Tournament selection
5          x̃ ⟵ Rand(𝒫)
6          x̃ ⟵ Evaluate(x̃)
7          if x̃ < x then
8              x ⟵ x̃
9              x ⟵ x̃
10     𝒮 ⟵ 𝒮 ∪ x
```

Algorithm 4.4: `SelectionOperator()` function

be found in the literature on the performance of different selection methods. For instance, it is expected that the random-walk technique performs worst, as there is no selection pressure, when "bad" individuals are as likely to be selected as "good" ones. The rank-based methods are expected to perform generally better than the fitness-proportional methods. Moreover, as Carter (1997) stated in his text, referring to the paper by Blickle and Thiele (1995), it has been claimed, on theoretical grounds, that tournament selection is the best method. Some other authors mention though that the fitness-proportional methods may be the best-working selection strategies for the core reload optimization (DeChaine and Feltus, 1995b). A more conclusive work would therefore be needed to find the best selection strategy for the problem.

The tournament selection was used for MYRRHA-FASTEF loading-pattern optimization. The function `SelectionOperator`$(\mathcal{P}, \nu, \mu)$ described in Algorithm 4.4 selects ν-times a random group of μ individuals from the population $\mathcal{P}$, each time it compares their fitness values[13] (i.e., runs the tournament), and picks the best individual (the winner). At the end, it returns a set $\mathcal{S}$ of ν parents. $\nu = 2$ and $\mu = 2$ were used in the calculations.

Crossover. Different mechanisms can lead to changes in the frequencies of alleles within a population of interbreeding organisms, which is an essential condition for evolution. Besides the selection mechanism described above, it is mainly the reproduction strategy implemented by means of different variation

[13]Note that the `Evaluate(`$\boldsymbol{x}$`)` function reads the LP characteristics from the memory, since all $\boldsymbol{x} \in \mathcal{P}$ have been already evaluated (cf. Algorithm 4.1 and 4.3).

operators. In the GA, two kinds of such operators are applied to evolve the population—crossover and mutation.

Crossover is the most important operator to carry the best genes in their genotype locations to next generations. It is a ν-ary operator analogous to biological mating, the process of creation of new individuals through reproduction. The role of the crossover operator is to recombine information from generally ν parent solutions into what is hoped to be even better offspring solutions (Poon and Carter, 1995). This is done by joining together gene sequences from two or more parent chromosomes, forming o new distinct offspring chromosomes, which may thereby give rise to or strengthen properties favored in the evolutionary race[14].

An ideal crossover operator should combine relevant characteristics of parents while producing valid (feasible) solutions to the problem, $\boldsymbol{x} \in \mathcal{F}$. However, when designing a crossover that generates viable offspring[15], one should be aware of the fact that "the offspring solution tends to receive more information from one parent and some randomization occurs. Both of these are generally considered bad," (Carter, 1997). In case unfeasible offspring are produced as a result of crossover, special repair mechanisms are usually employed that fix the solutions' chromosomes (see page 117). The form and efficiency of the crossover operator is directly related to the problem representation. An overview of crossover strategies for different types of representations can be found in Talbi (2009), for instance. Crossover operators suggested in literature for discrete ordering problems like the TSP or loading pattern optimization problem were reviewed and compared by Poon and Carter (1995). The following operators were included in the study[16]: Partially Mapped Crossover (PMX), Order Crossover (OX), Order Crossover #2 (OX2), Position Based Crossover (PBX), Cycle Crossover (CX), Tie-Breaking Crossover (TBX), Tie-Breaking Crossover #2 (TBX2), Intersection Crossover (IX), Union Crossover (UX), and Union Crossover #2 (UX2)[17]. Some of these operators have been applied to the loading pattern optimization problem; for instance, CX (Poon and Parks, 1992), PMX (Poon and Parks, 1992; Toshinsky et al., 1999), and some others like the Heuristic Copy & Match Crossover (HCMX) (Poon and Parks, 1992), one-point crossover (Do and Nguyen, 2007), two-point crossover (DeChaine and Feltus, 1995a, 1996; E. Tanker and A. Tanker, 1994; Ziver, Pain, et al., 2004), or the Heuristic Tie-Breaking Crossover (HTBX), which guarantees valid product solutions (Parks, 1996; Poon and Parks, 1993). An alternative approach was

[14] It is said that two genes cooperate, when some combinations of their (coadapted) alleles are beneficial. If a coadopted set of a parent is destructed being separated by crossover, one talks about disruption.

[15] E.g., using expert knowledge.

[16] This is an incomplete list.

[17] Poon and Carter (1995) claimed the UX2 to perform best for the TSP.

```
   Input  : p1, p2, N_FA, ζ
   Output: o1, o2

 1 o1 ∈ N_0^{N_FA}; o2 ∈ N_0^{N_FA}
 2 for i ⟵ N_FA do
 3 |   r ⟵ Rand(0, 1)                          // Crossover pattern
 4 |   if r < ζ then
 5 |   |   o1 i ⟵ p1 i
 6 |   |   o2 i ⟵ p2 i
 7 |   else
 8 |   |   o1 i ⟵ p2 i
 9 |   |   o2 i ⟵ p1 i
10 o1 ⟵ RepairOperator(o1)
11 o2 ⟵ RepairOperator(o2)
```

Algorithm 4.5: `CrossoverOperator()` function.

taken by Chapot et al. (1999), who avoided the use of heuristic operators by applying so-called List Model (LM). Boroushaki et al. (2003) in his GA copies only a part of the parent chromosomes and fills the rest randomly. Again, only a limited amount of information exists on the performance of different crossover operators used in the GAs for solving the loading pattern optimization problem. The rare work of Poon and Parks (1992) compared three crossover operators, CX, PMX, and HCMX, of which the last one gave clearly best results.

A typical design characteristic of many crossover operators applied to loading patterns is that they work with one-dimensional representations while disregarding the two-dimensional geometry of the loading pattern. Classical examples are basic n-point (-cut) crossovers that split parent chromosomes into $n+1$ chunks at n random loci, each of them covering a different region in the pattern, and exchange them. Shapes of these regions depend merely on the particular pattern-to-chromosome mapping and their number is substantially limited, as a consequence of the 2D-to-1D transformation, meaning that mutual combinations of some cooperating genes[18] are impossible. One way to face this problem is to perform a random walk through the two-dimensional position grid and select/exchange a region of adjacent positions in the core, which are more likely to carry coadopted alleles. This approach was implemented by DeChaine and Feltus (1996). Nevertheless, it is also possible that some disjoint

[18] See Footnote 14 on page 111.

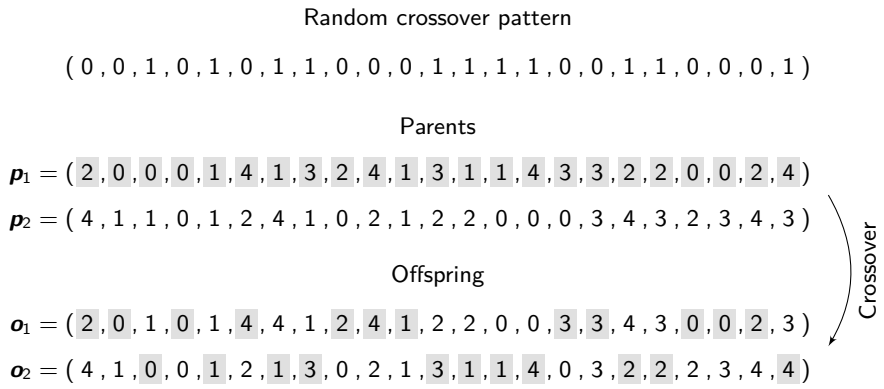


Figure 4.3: Uniform crossover

regions, not covered by this method, may also cooperate[19]. Therefore, it is suggested to use a simple uniform binary ($s = 2$) crossover operator that selects each parent gene to undergo crossover with the same probability ζ. Thus, exchange regions of arbitrary shapes and sizes are possible to be formed within the loading pattern, minimizing the unwanted bias introduced by other selection operators, but increasing the chance of the disruptive effect at the same time. The uniform operator with $\zeta = 0.5$ was used in this work, giving each gene the same probability to swap its allele or to keep it. The operator is described in Algorithm 4.5. A graphical demonstration of its application to two random parent chromosomes is shown in Figure 4.3.

Each reproduction operator is applied with a specified selection frequency, which is determined by the crossover rate $p_c \in [0, 1]$ in case of crossover. The best value of p_c depends on many GA design parameters such as the type of the selection procedure or the population size. The most commonly used rates are in the interval $[0.45, 0.95]$ (Talbi, 2009). Adaptive techniques for the crossover rate may also be useful, which allow to change its value during the optimization, $p_c \to p_c(g)$. The constant value of the crossover rate $p_c = 1$ was used in this work, causing the crossover operator to act on all sets of parents.

Mutation. Mutation is another reproduction operator in the GA. It is applied to every newborn offspring coming from the crossover with the probability $p_m \in [0, 1]$ (see Algorithm 4.1, line 19). It is introduced to maintain the

[19] In fact, the supremacy of the regional crossover method was not proved by DeChaine and Feltus (1996), as the observed improvement in the results against the two-point crossover stayed within the statistical error.

Input : $\boldsymbol{x}$, N_{FA}
Output: $\boldsymbol{x}$

```
1 T ⟵ {1, 2, ..., N_FA}
2 i ⟵ Rand(T)                                   // Mutation
3 j ⟵ Rand(T \ i)                               // pattern
4 Swap(x_i, x_j)
5 x ⟵ RepairOperator(x)
```

Algorithm 4.6: `MutationOperator()` function

diversity in the population $\mathcal{P}$ and thus to prevent premature convergence by making small random changes to a solution. In the biological metaphor, the operator simulates the role of the gene mutations in genetics.

Similarly to crossover, the mutation operator can also take various forms, mostly depending on the problem representation. Talbi (2009) reviews in his book the most prevalent concepts, while pointing out three important properties every good mutation operator should possess: ergodicity, meaning that every solution of the search space should be reachable through applying the operator; validity, a requirement on the production of valid solutions; and locality, that is that the mutation should cause only a minimal change. Basically the only type of the mutation operator used in the GAs for the loading-pattern optimization is the "binary exchange" or "random swap" mutation that randomly selects two genes in the solution chromosome and swaps their alleles (e.g., Alim, Ivanov, and S. H. Levine, 2008a; Toshinsky et al., 1999)[20]. The random-swap operator is described in Algorithm 4.6 and its principle is illustrated in Figure 4.4. This type of mutation operator is also implemented in the GA used in this work.

In general, mutation plays a less important role in GAs[21] and is applied with significantly lower frequencies than crossover; usually $p_{\mathrm{m}} \in [0.001, 0.010]$ (Talbi, 2009). DeChaine and Feltus (1995a) performed tests for fuel-swap probabilities 0 %, 10 %, and 50 %, each test consisting of 10 separate runs. It was concluded that the 0 % and 10 % tests were superior to the 50 % test and that the high 50 % factor was definitely detrimental to the optimization. However, it was hard to discern which of the first two values was better. The 10 % test finished with the highest fitness value, but the 0 % test climbed faster. That indicates that "some fuel swap mutation may be helpful, but the probability should be

[20] Three distinct mutation operators with heuristic rules designed for an advanced LP representation are described in Alim, 2006.

[21] Indeed, Poon and Parks (1992) revealed in their work, where they compared different crossover operators combined with mutation for different p_{c} and p_{m} values, that it is better to use a crossover operator than mutation alone.

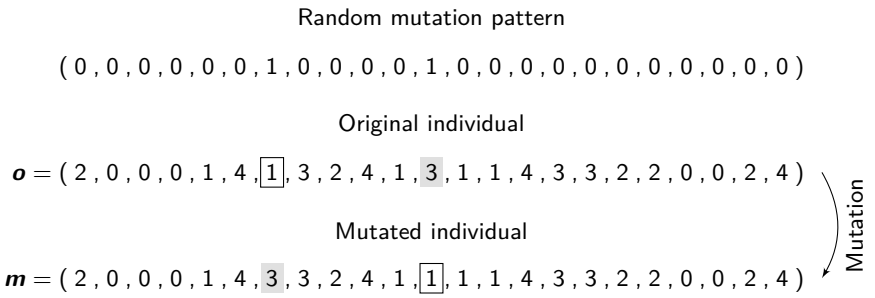


Figure 4.4: Random-swap mutation

kept small, for example, 10 % or less." Three values of the mutation rate were used here for the calculations: $p_\mathrm{m} \in \{0.00, 0.05, 0.20\}$ (e.g., Poon and Parks, 1992). It should be noted that likewise the crossover rate, the mutation rate can also take the form of a function, $p_\mathrm{m} \to p_\mathrm{m}(g)$.

As will be clarified later, the application of the mutation operator may also lead to infeasible loading patterns if higher core symmetries are employed. Therefore, the `RepairOperator()` is applied to every mutated solution at the end of the mutation algorithm.

Replacement strategy. Once fitness values for each of the offspring solutions have been calculated, a new population needs to be constructed from a combination of the previous population $\mathcal{P}$ and the children $\mathcal{O}$ (see Algorithm 4.1). There is a total of $|P \cup O| = p + o \times s$ solutions to choose from to create a population of a given constant size p. The principal replacement strategies discussed by Carter (1997) and Talbi (2009) are generational replacement (with or without elitism, see later) and steady-state replacement. In a pure generational replacement proposed by Holland (1975), the new population is made from the best of the $o \times s$ children produced and no member of the previous population is allowed to pass into it. In the steady-state replacement, only one offspring is generated and all members of the previous population (except the parents) are carried forward. The parents then compete, either deterministically or randomly, to complete the new population (Carter, 1997). An example of an alternative scheme is the population assessment technique used by Ziver, Pain, et al. (2004), which compares each offspring to a randomly chosen parent and replaces it if it is better. A distinct and commonly used concept that allows an "elite group" of size $\kappa \in [0, p-1]$ to be guaranteed survival from one generation to the next unaltered, is so-called elitism. When

```
   Input  : P, O, κ
   Output: Q

1  Q ⟵ ∅
2  if κ = 0 then Q ⟵ O                                   // No elitism
3  else                                                  // Elitism
4      T ⟵ P ∪ O; V ⟵ ∅
5      foreach x ∈ T do
6          x ⟵ Evaluate(x)
7          V ⟵ V ∪ x
8      v ⟵ Sort(V)                          // Sort in the ascending order
9      for i ⟵ 1 to κ do                     // Select κ best individuals
10         foreach x ∈ T do
11             x ⟵ Evaluate(x)
12             if x = v_i then
13                 Q ⟵ Q ∪ x
14                 Break()
```

Algorithm 4.7: `PopulationReplacement()` function

applied, it usually leads to faster but possibly premature convergence[22].

Two generational replacement strategies are used in this work; one without elitism ($\kappa = 0$, i.e., all parents are replaced by their children) and one with a slightly altered elitism, which passes $\kappa = p$ best individuals[23] from the merged population of parents and offspring $\mathcal{T}$ into the next generation $\mathcal{Q}$ (see the `PopulationReplacement()` function[24] in Algorithm 4.7).

Stopping criterion. Different stopping criteria may be used in the population-based methods, as the various arguments of the `StopCriterion()` function suggest in line 7 of Algorithm 4.1. Talbi (2009) distinguishes between two kinds of stopping criteria, depending on whether the end of the search is known a priory—static procedures, or not—adaptive procedures. An example of the static procedure is the `StopCriterion`(G, g) function from Algorithm 4.8, which terminates the search when a fixed number of iterations G is reached. Another example is the stopping criterion that limits the maximum number

[22] A phenomenon known in statistics as sampling error.

[23] In this case it is exactly one half of the merged-population size, since $s = p/2$ pairs ($\nu = 2$) of parents were used in all calculations and $o = 2$ children were produced per each pair of parents. Hence, $\kappa = o \times s = p$.

[24] See Footnote 13 on page 110.

Input : G, g
Output: a

1 $a \longleftarrow$ False
2 **if** $G = g$ **then** $a \longleftarrow$ True

Algorithm 4.8: `StopCriterion()` function

of objective function evaluations. The adaptive criteria are usually based on some statistics on the current population $\mathcal{P}$ or the evolution of the population. They quit the search, when the diversity measure falls below a given threshold; for instance, when the search stagnates and the best solution does not improve during a fixed number of successive iterations (Talbi, 2009). The stopping criterion from Algorithm 4.8 is used in this work. According to the literature, it is also the most frequently used criterion in the GAs for loading-pattern optimization.

Repairing strategy. Because infeasible individuals may be generated during population initialization or as a product of the crossover or mutation operators, special repair procedures are used to transform these into feasible solutions. Here, the infeasible solutions are those loading patterns that do not comply with the fuel-inventory constraints, that is, contain more fuel assemblies of a certain type than are available in the fuel stock. For example, let $\boldsymbol{m} = (15, 15, 15, 12, 12)$ be the fuel inventory vector and $\boldsymbol{t} = (0, 1, 2, 3, 4)$ the vector of corresponding fuel types for the loading pattern represented by the chromosome from Figure 4.2 in the one-third core symmetry. Then a feasible solution $\boldsymbol{x}$ must satisfy 5 equalities

$$\sum_{i=1}^{23} a_{i,j}\, s_i \leq m_j, \quad j = 1, 2, \ldots, 5\ , \tag{4.1}$$

where

$$a_{i,j} = \begin{cases} 1 & \text{if a core cell } i \text{ contains an assembly of type } t_j, \\ 0 & \text{otherwise ;} \end{cases} \tag{4.2}$$

and $\boldsymbol{s} = (s_1, s_2, \ldots, s_{23})$ are the cell symmetries (or frequencies), that is, the numbers of times individual cells appear in the full core. Different thinkable symmetries for the loading pattern from Figure 4.2 with the corresponding fuel-cell frequencies are shown in Figure 4.5[25]. Given the rules (4.1) and (4.2),

[25]Note that $s_i = 1$ for all cells, when a full core representation is considered without any symmetry. Also note that the middle and the right pattern from Figure 4.5 represent the same loading pattern, which is different from the loading pattern represented by the left pattern.

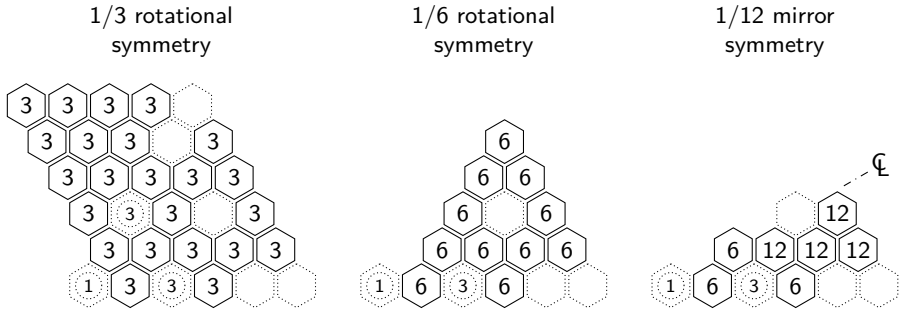


Figure 4.5: Loading pattern symmetries and cell frequencies

one can verify that the parent solutions $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ from Figure 4.3 are feasible, whereas the offspring solutions $\boldsymbol{o}_1$ and $\boldsymbol{o}_2$ are both infeasible.

The `RepairOperator()` function implemented in the GA 4.1 to fix the above described discrepancy is described in Algorithm 4.9. It is based on the ranking scheme used for the solution representation and allows to deal with the higher core symmetries from Figure 4.5. The function fills first the solution cells of the largest frequency value, then the cells of the second largest frequency value, and so on, until a feasible solution is completed. At each step, the cells are selected randomly from the group of cells of the same frequency that have not been selected yet. Every time a fuel-assembly type is to be assigned to a cell i of a frequency s, it is first checked, if the fuel type x_i assigned to the cell originally already satisfies the condition on its availability (see line 10 in the algorithm). If yes, s fuel assemblies of the corresponding type t_j are subtracted from the fuel stock m_j, the x_i value stays unchanged, and the algorithm proceeds with a next cell. If no, other value t_l is assigned to the cell, which is most similar to the original value x_i in terms of its rank, and for which enough fuel is available in the stock. In case two such values exist, one of them is selected randomly. Again, s units are subtracted from the stock m_l.

4.3.2 Ant Colony Optimization

ACO is another approximate population-based metaheuristic for solving hard combinatorial optimization problems. Like GA, ACO is also classified as an EA that progresses toward the optimum solution iteratively via improvements in the quality of a group (population) of solutions. However, unlike the local-search GA, which moves in the search space of complete solutions, ACO is a stochastic solution construction procedure, which works on partial solutions, trying to

Input : $\boldsymbol{x}, N_{\mathrm{FA}}, N, \boldsymbol{m}, \boldsymbol{t}, \boldsymbol{s}$
Output: $\boldsymbol{x}$

```
1  S ⟵ MakeSet(s)                                   // Unique frequencies
2  s̃ ⟵ SortReverse(S)                          // ...sorted in descending order
3  for s from s̃ do                          // Fill the high-frequency cells first
4      I ⟵ ∅
5      for i ⟵ 1 to N_FA do
6          if s_i = s then I ⟵ I ∪ i
7      while I ≠ ∅ do
8          i ⟵ Rand(I)
9          j ⟵ GetIndex(x_i, t)                                          // x_i = t_j
10         if s ≤ m_j then m_j ⟵ m_j − s
11         else
12             l ⟵ −1; d ∈ N_0^N                                // Distance vector
13             for k ⟵ 1 to N do d_k ⟵ |t_k − t_j|
14             D ⟵ MakeSet(d)                                 // Unique distances
15             d̃ ⟵ Sort(D \ 0)                          // ...sorted in ascending order
16             for d from d̃ do
17                 L ⟵ GetIndexes(d, d)                  // Closest ranks indexes
18                 while L ≠ ∅ do                                       // |L| ∈ {1, 2}
19                     u ⟵ Rand(L)
20                     if s ≤ m_u then
21                         l ⟵ u
22                         Break()
23                     else L ⟵ L \ l
24                 if l ≠ −1 then
25                     x_i ⟵ t_l
26                     m_l ⟵ m_l − s
27                     Break()
28         I ⟵ I \ i
```

Algorithm 4.9: `RepairOperator()` function

extend these in the best possible way to complete problem solutions (Dorigo and Stützle, 2010). The step-by-step procedure of constructing the solutions will be described in detail later in this section. More specifically, ACO is classified as a "swarm-intelligence" technique. Boussaïd et al. (2013) describe SI as "an innovative distributed intelligent paradigm for solving optimization problems that takes inspiration from the collective behavior of a group of social insect colonies and of other animal societies. SI systems are typically made up of a population of simple agents (an entity capable of performing/executing certain operations) interacting locally with one another and with their environment. These entities with very limited individual capability can jointly (cooperatively) perform many complex tasks necessary for their survival. Although there is normally no centralized control structure dictating how individual agents should behave, local interactions between such agents often lead to the emergence of global and self-organized behavior." Examples of the social insects imitated by the SI techniques are bees, wasps, termites, and other animal societies such as flocks of birds or fish schools. Clearly, the social insects in ACO are ants.

ACO was developed and formalized by Marco Dorigo and his colleagues in 1990s (e.g.. Dorigo, Maniezzo, et al., 1996; Dorigo, Di Caro, et al., 1999). Its main inspiration comes from real ants' foraging behavior, which is based on the stigmergy, a kind of indirect communication between the ants by means of trails of a chemical substance called pheromone. When searching for food, ants prefer to follow directions rich in pheromone, a certain amount of which is deposited on the ground by every walking ant. Since the pheromone evaporates everywhere at a constant rate, its amount decreases in time and shorter paths become to be preferred by ants, as they find more pheromone on them on their way back to the nest. This enables the ants to discover the environment and adapt to changes in it, in case old paths are no longer feasible or are feasible again, due to newly appearing/disappearing physical obstacles. Additionally, it also allows the ants to find shortest paths between their nest and food sources. Several ACO algorithms have been proposed that utilize these principles. They differ mostly in the particular implementation of the rules for pheromone evaporation and its deposition (reinforcement) and include methods like Ant System algorithm (AS), Quantum Ant Colony Optimization (QACO), Rank-based Ant System (RAS), Best-Worst Ant System (BWAS), Ant Colony System (ACS), or Max-Min Ant System (MMAS). More information regarding the history and recent trends in the ACO computing, theoretical work concerning the algorithm convergence, various fields of its application, and closer review of the different ACO methods are given in the surveys Blum, 2005; Dorigo and Stützle, 2010.

ACO shares with GA basically the same characteristics regarding its applicability as a black-box method for solving different optimization problems and its optimization performance (see Section 4.3.1). Being a population-based method,

ACO is equally well amenable to efficient parallelization and the special way of constructing solutions makes it very easy to implement local optimization heuristic rules. Like in case of GA, combination with a local search procedure is recommended to improve algorithm's efficiency.

One of the first ACO applications for solving the loading pattern optimization problem was done by L. Machado and Schirru, who used the Ant-Q algorithm to optimize a PWR core reload. More applications followed later, using various ACO versions (cf. Table A.1): ACS was used by De Lima, Schirru, et al. (2008), MMAS by Hoareau (2008), RAS by Wang and C. Lin (2009), and QACO by Silva, Schirru, and Lima (2011). Esquivel-Estrada et al. (2011) compared four ACO methods in their paper on solving a BWR reload problem—ACS, AS, BWAS, and MMAS. It was concluded that the BWAS was the best performing algorithm among them (followed closely by the MMAS) and, also, that it performed better than GA, neural networks, and TS. In a similar article by C. Lin and B.-F. Lin (2012), RAS, MMAS, and Ant-Q were compared, when applied to solve a PWR loading pattern optimization problem. In this case, the MMAS had a better performance than the other methods. L. Machado and Schirru compared his Ant-Q to a GA, but did not make any definite conclusions on the superiority of one of the algorithms over the other. Another more comprehensive comparison involving also some other metaheuristics was already mentioned in Section 4.3.1 in Footnote 1.

Algorithm

A basic ACO algorithm consists of three essential parts—initialization, construction of ant-based solutions, and pheromone update. During initialization, a set $\mathcal{T}$ is defined that includes pheromone values $\tau \in \mathcal{T}$ associated with all solution components. This set represents a parametrized probabilistic model commonly known as the pheromone model (Blum, 2005). With $\mathcal{T}$ defined, the ACO approach solves an optimization problem by iterating two steps: in the first one, a set of k artificial ants build stochastically and incrementally solutions $\boldsymbol{x}$ to the considered problem, using the pheromone model $\mathcal{T}$ and heuristic information $\mathcal{H}$; and then, in the second step, the constructed solutions $\mathcal{A}$ are used "to modify $\mathcal{T}$ in a way that is deemed to bias future sampling toward high quality solutions," (Blum, 2005). The first step is then repeated again with the updated $\mathcal{T}$ and the two-step iterative scheme proceeds in the same fashion until a given stopping criterion is satisfied.

The ACO used for MYRRHA-FASTEF loading-pattern optimization is depicted in Algorithm 4.10. This algorithm was mainly inspired by the ACS and Ant-Q algorithms proposed by De Lima, Schirru, et al. (2008) and by L. Machado and

```
Input  : I, N_FA, N, H, τ0, t, m, s, k, q0, α, β, ρ, γ
Output: x_opt, x_opt

1  i ⟵ 0; x_opt ⟵ −∞
2  foreach τ_rsp ∈ T do τ_rsp ⟵ τ0                          // Initialization
3  while StopCriterion(I, i) is False do                    // Stopping criterion
4      i ⟵ i + 1; A ⟵ ∅
5      for j ⟵ 1 to k do
6          x ⟵ ConstrAntSol(T, H, N_FA, N, t, m, s, q0, α, β)   // Solution
7          x ⟵ Evaluate(x)                                  // construction
8          if x > x_opt then
9              x_opt ⟵ x
10             x_opt ⟵ x
11         A ⟵ A ∪ x
12     T ⟵ UpdatePheromone(N_FA, T, A, τ0, ρ, γ)            // Pheromone update
```

Algorithm 4.10: Ant Colony Optimization

Schirru (2002), respectively, to solve analogous problems. A detailed description of the algorithm's components is given in the following paragraphs.

Initialization. All pheromone values $\tau_{r\,s\,p} \in \mathcal{T}$ are set to the initial user-specified value τ_0 at the beginning of the algorithm. The meaning of the indexes r, s, and p is explained in the next paragraph, as it is related to the way the solutions are constructed in the ACO. $\tau_0 = 1$ was used in this work.

Construction of ant solutions. The process of constructing problem solutions $\boldsymbol{x}$ is one of the defining characteristics of the ACO algorithms. It usually has different forms for different problems and sometimes, alternative ways of constructing solutions may be considered even for the same problem. This is also the case of the ACOs used for solving the loading pattern optimization problem. For instance, the ants in the ACO implemented by Esquivel-Estrada et al. (2011) move in a randomly generated loading pattern from one random position ("source channel") to a different position ("destination channel") by applying a probabilistic formula that will be called the state transition rule (STR). Then, when the ants stop their movement, the fuel assemblies in the source channels and destination channels change positions between them and new loading patterns (solutions) are obtained. Hoareau (2008) builds the loading patterns in his article using ants that choose randomly an assembly family and decide to place it at a position, applying the STR. The strategy adopted in this

work was used first by L. Machado and Schirru (2002) and later by some other authors (e.g., De Lima, Schirru, et al., 2008).

In this strategy, an artificial ant is an agent which moves through all fuel-assembly positions in a loading pattern in a given order and every time it visits a new position, it chooses which fuel-assembly type will be loaded into the next position. In the insect analogy, the transition from the initial position to the final position corresponds to the transition from the nest to the food source. However, while real ants take different paths when searching for food localized on a flat surface, all artificial ants take an identical path when walking through the MYRRHA-FASTEF loading pattern, but their altitude can change in discrete values with every new position, which corresponds to different fuel-assembly types. Thus, the artificial ant moves in a three-dimensional landscape rather than in the two-dimensional landscape of the real ant[26]. In the present algorithm, an ant starts its journey in the central core position that corresponds to cell 1 from Figure 4.2 (see the cell locus numbers) and it then continues through cells 2, 3, and so on, until it closes its tour by reaching the final cell 23.

In order to choose, which of the available solution components should be added to the current partial solution $\boldsymbol{x}_p = (x_1, x_2, \ldots, x_p)$ for $p = 1, 2, \ldots, N_{\mathrm{FA}} - 1$, a probabilistic decision is made by the ant, using the state transition rule (STR)

$$s = \begin{cases} \underset{s}{\mathrm{argmax}} w(r, s, p) & \text{if } q \leq q_0 \ , \\ \text{roulette} & \text{if } q > q_0 \ , \end{cases} \tag{4.3}$$

where

$$w(r, s, p) = \tau_{r\,s\,p}^{\alpha} \times \eta_{r\,s\,p}^{\beta}, \quad s \in \mathcal{U}_{r\,p} \ . \tag{4.4}$$

In Eq. (4.3) and (4.4), r is the current solution component at the position p ($r = x_p$), s is the next solution component at the position $p + 1$ ($s = x_{p+1}$), $\tau_{r\,s\,p} \in \mathcal{T}$ is the pheromone trail on the arc (r, s, p), $\eta_{r\,s\,p} \in \mathcal{H}$ is possible heuristic information associated to the ant's move (r, s, p), $q \in [0, 1]$ is a random number, and $q_0 \in [0, 1]$, α, and β are parameters. In the current problem, the solution components are the fuel types $\boldsymbol{t}$. Their availability has to be checked against the fuel inventory $\boldsymbol{m}$ whenever the partial solution is to be extended. When a fuel assembly of the type t_i is appended to the solution staying at a position p, s_{p+1} items have to be removed from the inventory, $m_i \leftarrow m_i - s_{p+1}$, where $\boldsymbol{s}$ is the vector of the fuel-cell symmetries from Figure 4.5. Therefore, it is necessary for the ants to perform these actions that they are endowed with a special working memory used to memorize which types of the fuel assemblies and how many of them have been used so far or, respectively, are still available.

[26]The two-dimensional moves of ants are preserved in ACO algorithms for solving TSP-like problems, for instance.

The corresponding set of all possible moves from position p loaded with a fuel assembly of the type r is denoted by $\mathcal{U}_{r\,p}$.

The "roulette" from Eq. (4.3) is defined as a random proportional rule with a probability distribution $P(r, s, p)$ that is a function of $\tau_{r\,s\,p}$ and $\eta_{r\,s\,p}$:

$$P(r,s,p) = \frac{\tau_{r\,s\,p}^{\alpha} \times \eta_{r\,s\,p}^{\beta}}{\sum_{u \in \mathcal{U}_{r\,p}} \tau_{r\,u\,p}^{\alpha} \times \eta_{r\,u\,p}^{\beta}} \ . \tag{4.5}$$

The use the of local heuristics $\eta_{r\,s\,p}$ in the above formulas enables to employ the problem-specific information during the search, which may improve the algorithm's performance and guide the ants to better results. An example of a local heuristic when searching for a flat power distribution in the reactor core may be that only fuel assemblies with similar burnup can be placed in two neighboring locations. Also, $\eta_{r\,s\,p}$ may be set to zero when a fuel assembly of a certain type s is forbidden to be inserted into a position $p + 1$. Similar and some other heuristics were used by many authors in their works; for instance, by De Lima, Schirru, et al. (2008), Esquivel-Estrada et al. (2011), Hoareau (2008), L. Machado and Schirru (2002), and Wang and C. Lin (2009). A special dedicated study on heuristics in an ant system for nuclear reload optimization was prepared by De Lima, M. D. Machado, et al. (2007). Heuristic information may also be very useful at the initial phase of the algorithm, as it can attract ants to preferred paths, when the same initial amount of pheromone is deposited everywhere in the search graph. When defining the set $\mathcal{H}$ of heuristic rules for the present version of ACO, one has to take into account the particular route the ants are taking through the loading pattern. This may lead to the same problems as those encountered when designing a good GA crossover operator for a given problem representation (see Section 4.3.1). In this work, no local heuristics are used. Accordingly, $\alpha = 1$ and $\beta = 0$ in the calculations, as both parameters determine the relative respective influence of the pheromone values and the heuristic values on the decision of the ant.

The rules (4.3)–(4.5) introduce a biased exploration toward arcs marked with more pheromones, while maintaining some level of stochasticity, determined by the parameter q_0. Values of this parameter are usually set high, so the ants choose mostly for the paths with higher pheromone concentrations. Two values $q_0 = 0.8$ and $q_0 = 0.9$ were considered for the MYRRHA loading-pattern optimization.

The complete algorithm, which describes the process of constructing ant solutions in the ACO applied in this work, is depicted in Algorithm 4.11. Note that a special part of the algorithm is followed by an ant, when it makes its first move (lines 5–27), and other special part, when it makes all subsequent moves (lines 28–45).

Input : $\mathcal{T}, \mathcal{H}, N_{\text{FA}}, N, \boldsymbol{t}, \boldsymbol{m}, \boldsymbol{s}, q_0, \alpha, \beta$
Output: $\boldsymbol{x}$

$\boldsymbol{x} \in \mathbf{N}_0^{N_{\text{FA}}}$
for $p \longleftarrow 1$ **to** $N_{FA} - 1$ **do**
  $s \longleftarrow -1$
  $q \longleftarrow \texttt{Rand}(0, 1)$
  if $p = 1$ **then** // Construct arc 1
    $\mathcal{U}_p \longleftarrow \emptyset$ // Set of possible arcs (r, s)
    $\boldsymbol{v} \longleftarrow \mathbf{N}_0^2$
    for $i \longleftarrow 1$ **to** N **do**
      for $j \longleftarrow 1$ **to** N **do**
        if $(i = j$ **and** $m_i - s_p - s_{p+1} \geq 0)$ **or** $(i \neq j$ **and** $m_i - s_p \geq 0$ **and** $m_j - s_{p+1} \geq 0)$ **then** $\mathcal{U}_p \longleftarrow \mathcal{U}_p \cup (t_i, t_j)$
    if $q < q_0$ **then** // State transition rule
      $\mathcal{V} \longleftarrow \texttt{ArgMax}(\{w(u_1, u_2, p) | \boldsymbol{u} \in \mathcal{U}_p\})$ // See Eq. (4.4)
      $\boldsymbol{v} \longleftarrow \texttt{Rand}(\mathcal{V})$
    else
      $a \longleftarrow 0$
      $t \longleftarrow \texttt{Sum}(\{\tau^{\alpha}_{u_1\, u_2\, p} \times \eta^{\beta}_{u_1\, u_2\, p} | \boldsymbol{u} \in \mathcal{U}_p\})$ // $\tau \in \mathcal{T}, \eta \in \mathcal{H}$
      $\tilde{\boldsymbol{u}} \longleftarrow \texttt{MakeArray}(\mathcal{U}_p)$ // Unsorted array of arrays
      $\tilde{q} \longleftarrow \texttt{Rand}(0, 1)$
      for $\boldsymbol{u}$ **from** $\tilde{\boldsymbol{u}}$ **do**
        $a \longleftarrow a + \tau^{\alpha}_{u_1\, u_2\, p} \times \eta^{\beta}_{u_1\, u_2\, p} \times t^{-1}$
        if $a > \tilde{q}$ **then** // Roulette
          $\boldsymbol{v} \longleftarrow \boldsymbol{u}$
          $\texttt{Break}()$
    $x_p \longleftarrow v_1$ // $v_1 = r$
    $i \longleftarrow \texttt{GetIndex}(v_1, \boldsymbol{t})$
    $m_i \longleftarrow m_i - s_p$
    $s \longleftarrow v_2$
  else // Construct arcs $2, 3, \ldots, N_{\text{FA}} - 1$
    $\mathcal{U}_{r\, p} \longleftarrow \emptyset$ // Set of possible destinations s
    $r \longleftarrow x_p$
    for $i \longleftarrow 1$ **to** N **do**
      if $m_i - s_{p+1} \geq 0$ **then** $\mathcal{U}_{r\, p} \longleftarrow \mathcal{U}_{r\, p} \cup t_i$
    if $q < q_0$ **then** // State transition rule
      $\mathcal{V} \longleftarrow \texttt{ArgMax}(\{w(r, u, p) | u \in \mathcal{U}_{r\, p}\})$ // See Eq. (4.4)
      $s \longleftarrow \texttt{Rand}(\mathcal{V})$
    else
      $a \longleftarrow 0$

Algorithm 4.11: `ConstrAntSol()` function

```
38          ũ ⟵ MakeArray(U_rp)          // Unsorted array of integers
39          t ⟵ Sum({τ^α_rup × η^β_rup | u ∈ U_rp})
40          q̃ ⟵ Rand(0, 1)
41          for u from ũ do
42              a ⟵ a + τ^α_rup × η^β_rup × t^−1
43              if a > q̃ then                          // Roulette
44                  s ⟵ u
45                  Break()
46      x_{p+1} ⟵ s
47      i ⟵ GetIndex(s, t)
48      m_i ⟵ m_i − s_{p+1}
```

Algorithm 4.11: `ConstrAntSol()` function (continuation)

Solution evaluation. The same function `Evaluate()` was used in the ACO as the one used in the GA from Section 4.3.1 (Algorithm 4.1). It is depicted in Algorithm 4.3. The only difference is that the iteration number g, used in the GA, is denoted by the letter i in the ACO.

Pheromone update. The role of the pheromone values $\tau \in \mathcal{T}$ in ACO is to mediate cooperation between ants. In order to concentrate the search in regions containing high-quality solutions, these values are dynamically changed twice during each iteration; first, by means of so-called local updating rule, and second, by means of so-called global updating rule.

In the current ACO, the local updating rule is applied uniformly to all pheromone values as follows:

$$\tau_{r\,s\,p} \longleftarrow (1-\rho)\,\tau_{r\,s\,p} + \rho\,\tau_0, \quad \tau_{r\,s\,p} \in \mathcal{T}\ . \tag{4.6}$$

Here, the first summand on the right-hand side reflects what is known from nature as pheromone evaporation and can be interpreted as an ACO-specific implementation of the concept of forgetting. The goal of the pheromone evaporation is to make the solution components less and less attractive as they are not visited by ants, which helps to avoid a premature convergence of the algorithm and allows the ants to explore not yet visited areas of the search space. The intensity of the pheromone evaporation, which decreases all pheromone values over time, is determined by the evaporation rate ρ. The second summand in the local updating rule (4.6) is introduced to ensure a minimum level of exploration of all areas by maintaining a minimum non-zero pheromone value

```
Input  : N_FA, T, A, τ_0, ρ, γ
Output: T

x_iter ⟵ −∞
foreach x ∈ A do
    x ⟵ Evaluate(x)
    if x > x_iter then
        x_iter ⟵ x
        x_iter ⟵ x                                  // Iteration-best solution
foreach τ_r s p ∈ T do
    τ_r s p ⟵ (1 − ρ) τ_r s p + ρ τ_0                // Local updating rule
for i ⟵ 1 to N_FA − 1 do
    j ⟵ x_iter i
    k ⟵ x_iter i+1
    foreach τ_r s p ∈ T do
        if r = j and s = k and p = i then
            τ_r s p ⟵ (1 − ρ) τ_r s p + ρ γ x_iter   // Global updating rule
```

Algorithm 4.12: `UpdatePheromone()` function

for all solution components. In this particular case, $\tau_{r\,s\,p} \geq \tau_0$ for all $\tau_{r\,s\,p} \in \mathcal{T}$ at every stage of optimization.

The global updating rule is intended to reward solution components belonging to better solutions and make these more attractive for ants in the following iterations. It is applied after all ants have constructed a complete solution and the solution $\boldsymbol{x}$ has been evaluated by an objective function f. One can think of the value $f(\boldsymbol{x})$ as the quality of the food source in the real ant world. Several instantiations of the global updating rule have been proposed for ACO, depending on which solutions were used for the update to reflect the acquired search experience (e.g., Blum, 2005; Talbi, 2009). In the present work, the global updating rule gets the form

$$\begin{aligned} \tau_{r\,s\,p} &\longleftarrow (1-\rho)\,\tau_{r\,s\,p} + \rho\,\gamma\,f(\boldsymbol{x}_{\mathrm{iter}})\ , \\ r = x_{\mathrm{iter}\,p}, \quad s &= x_{\mathrm{iter}\,p+1}, \quad p = 1, 2, \ldots, N_{\mathrm{FA}} - 1\ , \end{aligned} \tag{4.7}$$

where $\boldsymbol{x}_{\mathrm{iter}}$ is the best solution in the respective iteration and γ is a parameter[27]. The right-most product in the formula (4.7) is considered the reinforcement

[27] Other frequently used form of the global updating rule uses the best solution found so far rather than the iteration-best solution. One then usually talks about the elitist pheromone update (Talbi, 2009).

learning term of the algorithm.

The pheromone update is schematically depicted in Algorithm 4.12. Two values were used in this work for the pheromone evaporation rate, $\rho = 0.1$ and $\rho = 0.2$.

Stopping criterion. The maximum iteration-number stopping criterion from Algorithm 4.8 was applied in the ACO, with the letters I and i used instead of G and g, respectively.

4.4 Optimization Problems

Two hypothetical MYRRHA-FASTEF loading pattern optimization problems are solved in this section, using the GA and ACO algorithms described in the preceding section. Both problems were designed with different intentions; the first one, Problem 1, as a test/learning problem, and the second one, Problem 2, as an example of a more realistic fuel-management problem. The same MYRRHA-FASTEF critical-core geometry with the one-third core symmetry (see Figure 2.4) is assumed in both cases with the same constraints on the available fuel.

4.4.1 Problem 1

Problem 1 is a simple problem with a known optimum solution. Solving this problem allows testing the applied optimization methods, find appropriate ranges for the principal solution parameters, and make some conclusions on the methods' performances.

Problem Definition

Problem 1 is defined as follows:

$$\underset{\boldsymbol{x}}{\text{Maximize}} \quad k_{\text{eff}}(\boldsymbol{x}) \ , \tag{4.8}$$

$$\text{subject to} \quad \sum_{i=1}^{N_{\text{FA}}} \delta_{x_i\, t_j} \leq m_j, \quad j = 1, 2, \ldots, N \ , \tag{4.9}$$

where $\boldsymbol{x} = (x_1, x_2, \ldots, x_{N_{\text{FA}}})$ is the loading pattern, $k_{\text{eff}}(\boldsymbol{x})$ is the (BOC) effective multiplication factor, $\delta_{i\,j}$ is the Kronecker delta, $N_{\text{FA}} = 23$ is the number of fuel

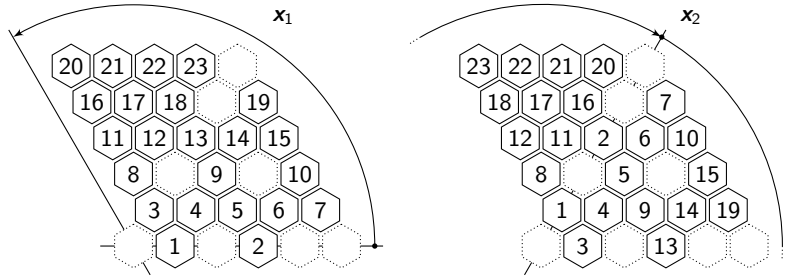


Figure 4.6: Equivalent solutions

assemblies in one third of the core, $N = 5$ is the number of different considered fuel types (ranks) $\boldsymbol{t} = (0, 1, 2, 3, 4)$, and $\boldsymbol{m} = (5, 5, 5, 4, 4)$ is the corresponding fuel availability, again considering one third of the core only[28].

Solution Methods

The GA from Section 4.3.1 and ACO from Section 4.3.2 were used to solve the problem with approximately 2.6×10^{13} unique feasible solutions $\boldsymbol{x} \in \mathcal{F}$ (see Section 2.2.3)[29]. The components and parameter settings of both implemented algorithms are summarized in Table 4.1 and Table 4.2. As was explained earlier in the previous sections, the values of the parameters and their ranges were carefully chosen based on an extensive literature search. While a good agreement on some parameter values exists for the given problem, somewhat inconsistent or not very accurate information on some other parameters can also prevail. Therefore, Problem 1 was solved for several values of such parameters to identify which of them give good results within a reasonable computation time. This simple parametric study should thus provide sufficiently efficient and robust optimization algorithms for solving the more realistic Problem 2. It is admitted that a more complex study including more parameters or application of some more intelligent parameter-tuning techniques[30] would be needed to obtain highly optimized algorithms. However, this work would go beyond the scope of the present dissertation and, therefore, has not been done. An "offline" parameter initialization was used in case of both algorithms; that is,

[28]Note that the symmetry of all FA cells is three-fold in this case $s_i = 3$, (see Figure 4.5).

[29]It should be noted that the number of unique objective-function values equals one half of this value only, $|\mathcal{F}| = 2 \times |\{f(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{F}\}|$, since for each LP $\boldsymbol{x}_1$ a different LP $\boldsymbol{x}_2$ exists with the same objective function value: $f(\boldsymbol{x}_1) = f(\boldsymbol{x}_2)$. An example of two such equivalent LPs is shown in Figure 4.6; both LPs represent the same core configurations if the same fuel is loaded into the cells denoted by the same numbers.

[30]E.g., Birattari, 2009.

Table 4.1: GA settings

Component	Implemented variant, parameter value
Problem representation	(rank-based) integer-string representation
Initial population	randomly generated population of size $p \in \{30, 50, 80\}$
Selection operator	tournament selection of size $\mu = 2$
Crossover operator	binary uniform crossover with the swap probability $\zeta = 0.5$ and crossover rate $p_c = 1.0$
Mutation operator	binary random-swap mutation with the mutation rate $p_m \in \{0.00, 0.05, 0.20\}$
Replacement strategy	generational replacement with ($\kappa = p$) or without ($\kappa = 0$) elitism
Stopping criterion	maximum number of generations $G = \{100, 200\}$[a]

[a] Note that $g = 0$ corresponds to the initial population.

Table 4.2: ACO settings

Component	Implemented variant, parameter value
Initial pheromone amount	$\tau_0 = 1$
Number of ants	$k \in \{30, 50, 80\}$
Evaporation rate	$\rho \in \{0.1, 0.2\}$
STR parameters	$q_0 \in \{0.8, 0.9\}, \alpha = 1, \beta = 0$
Stopping criterion	maximum number of iterations $I \in \{200, 268\}$[a]

[a] Note that the optimization starts with the iteration number $i = 0$.

the parameters were given fixed values at the beginning and stayed constant till the end of the execution[31].

Each optimization case characterized by a given set of initial parameters was executed 10 times, each independent run starting with a different random-number generator seed. The optimization was done using the optimization framework from Chapter 5. The simplified diffusion model of the MYRRHA-FASTEF reactor core described in Section 3.3.1 and characterized by $P = 100\,\text{MW}$, $N_g = 15$, $R = 0$, and $\beta = 0.852$ was used in both algorithms with DIF3D10.0 as the reactor analysis code. All calculations were performed in a Linux environment on a Dell Latitude E6510 laptop with an Intel® Core™ i5 M520 processor ($4 \times 2.4\,\text{GHz}$) and 3.7 GiB of RAM, employing a single CPU in a serial mode.

[31] In contrast to the "online" parameter initialization that allows for a continuous change in the parameter values during the optimization (e.g., Talbi, 2009).

GA Results

The results of the optimization of Problem 1 using the GA are summarized in Figure 4.7 and Table 4.3. 18 different GA settings were used in total to solve the problem, corresponding to all combinations of three different population sizes $p \in \{30, 50, 80\}$, three mutation rates $p_\mathrm{m} \in \{0.00, 0.05, 0.20\}$, and two generational replacement strategies (with and without elitism).

Several observations were made when analyzing the results. Admittedly, the most noticeable one is the impact of employing elitism on the solution convergence and the success rate in finding the known global optimum solution, which is depicted together with the loading pattern with the minimum k_eff in Figure 4.8 and characterized by $k_\mathrm{eff} = 1.016600$. It can be clearly seen from the results that the elitist strategy led almost always to the optimum solution. The only exceptions were two $p = 30$ cases, for which the optimum solution was not found twice ($p_\mathrm{m} = 0.00$), respectively, once ($p_\mathrm{m} = 0.05$) out of ten trials, and that due to the premature convergence. It can also be seen that the optimum solution was found at a fairly early stage of the optimization, on average between the 20th and 40th generation. The consistency with which it was found grows with p but is relatively insensitive to p_m. For example, the optimum was always identified between the 16th and 26th generation for $p = 80$ for all mutation rates, whereas for $p = 50$ this range was between 18 and 66 and for $p = 30$, when the optimization was successful, between 19 and 127.

Generally worse results were obtained and different patterns were observed in the results when elitism was not employed. For instance, the highest success rates in finding the optimum solution were achieved for $p = 30$ and the lowest ones for $p = 80$. Actually, the optimum solution was not found a single time for $p = 80$ and $p_\mathrm{m} = 0.20$, whilst it was found six times for $p = 30$ and $p_\mathrm{m} = 0.00$. Moreover, as Figure 4.7 shows, the population-best solution $k^*_\mathrm{eff}(g)$ kept steadily improving throughout the whole optimization only for $p = 30$ and all p_m values, and no obvious continuous improvement was observed for larger populations. Instead, it seems that the $k^*_\mathrm{eff}(g)$ values for $p = 50$ and $p = 80$ began to stagnate after the ≈ 50th generation and to fluctuate around values that are typically higher for lower p_m. The fact that the fitness stopped to improve and did not evolve in the later stages of the optimization is further illustrated in Figure 4.9–4.11, which show the progression of the $k^*_\mathrm{eff}(g)$ distributions through the generations $g = 0$ (random initial population), 10, 100, and 200 for different population sizes. These observations are in correspondence with those made by Poon and Parks (1992), who explained them by "the disruptive nature of the crossover operator, which, while being initially beneficial, enabling good areas of the search space to be located quickly, becomes less useful as a run proceeds." The authors also added that "surprisingly, neither increasing the

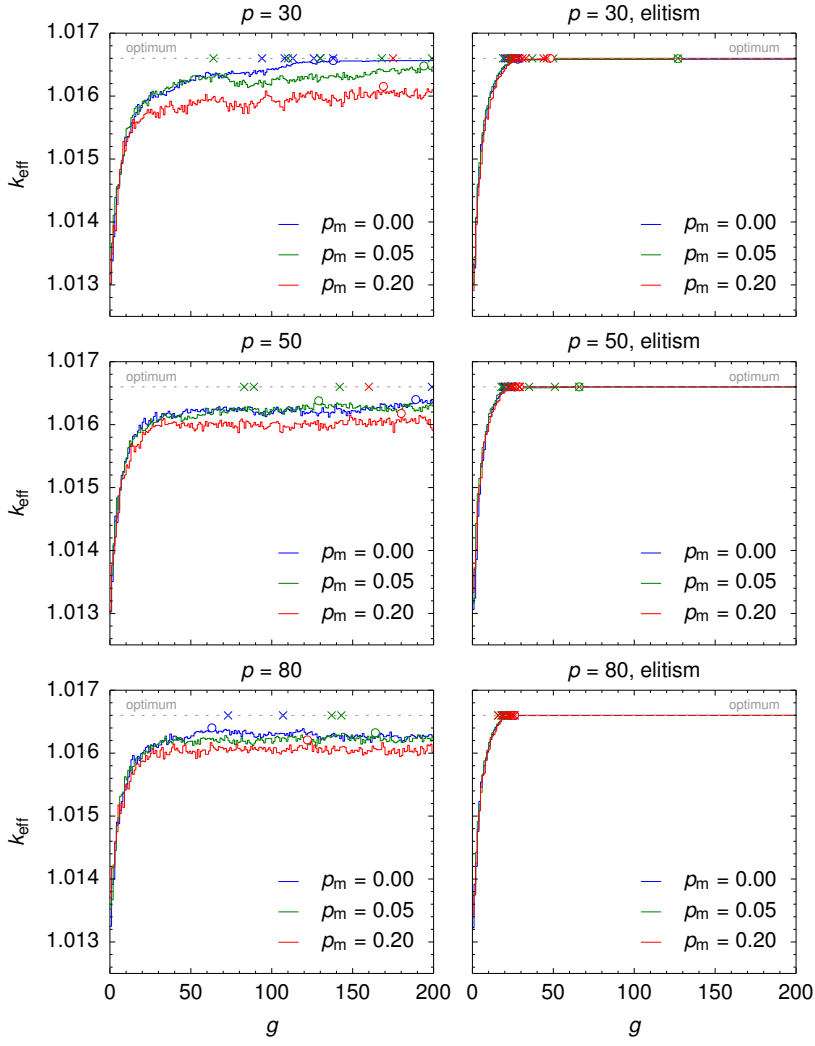


Figure 4.7: Problem 1 results (GA): evolution of the population-best fitness value (k_{eff}) obtained for different population sizes p and mutation rates p_{m} and with or without elitism; the values were obtained by averaging results of 10 independent runs; "×" indicates the first time (if) the optimum solution was found for each of the 10 runs; "○" indicates the first time the maximum average value was achieved

Table 4.3: Problem 1 results (GA)

#	GA parameters			k^*_{eff} (fitness)		Optimum identification				ANE[a]
	p	p_{m}	Elitism	Avg.	Max.	Count (#/10)	Avg. g	Min. g	Max. g	
1	30	0.00	no	1.016571	1.016600	6	118	94	138	3121
2	30	0.05	no	1.016571	1.016600	5	134	64	199	4743
3	30	0.20	no	1.016444	1.016600	1	175	175	175	5784
4	50	0.00	no	1.016552	1.016600	1	199	199	199	9785
5	50	0.05	no	1.016551	1.016600	2	86	83	89	9367
6	50	0.20	no	1.016495	1.016600	1	160	160	160	9952
7	80	0.00	no	1.016567	1.016600	1	73	73	73	15850
8	80	0.05	no	1.016547	1.016600	2	140	137	143	15900
9	80	0.20	no	1.016471	1.016556	0	–	–	–	15956
10	30	0.00	yes	1.016587	1.016600	8	24	20	28	791
11	30	0.05	yes	1.016593	1.016600	9	39	19	127	918
12	30	0.20	yes	1.016600	1.016600	10	31	22	46	1200
13	50	0.00	yes	1.016600	1.016600	10	22	19	25	1195
14	50	0.05	yes	1.016600	1.016600	10	30	18	66	1370
15	50	0.20	yes	1.016600	1.016600	10	26	21	29	1718
16	80	0.00	yes	1.016600	1.016600	10	21	19	26	1935
17	80	0.05	yes	1.016600	1.016600	10	20	16	23	2131
18	80	0.20	yes	1.016600	1.016600	10	22	16	26	2300

[a] Average number of evaluations.

The values in each row were obtained by processing results of 10 independent runs.

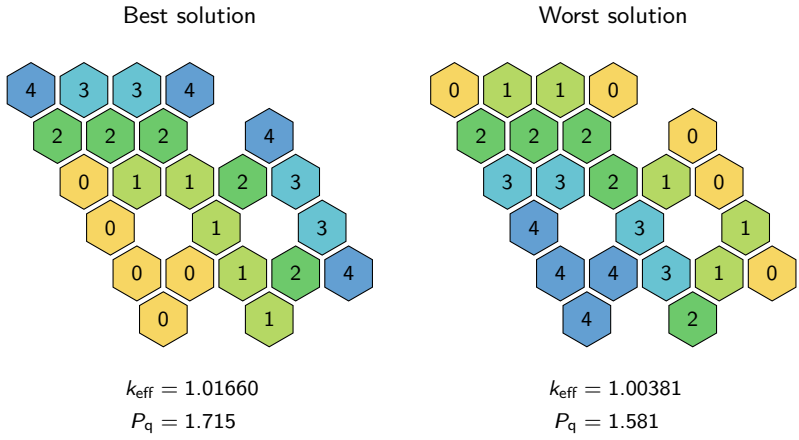


Figure 4.8: Problem 1 best and worst solution

population size nor increasing the mutation rate and selection pressure has alleviated this problem." Indeed, moving from $p = 50$ to $p = 80$ did not lead to any improvement in the algorithm convergence, but rather vice versa, as follows from Figure 4.7. Also, as anticipated, this behavior becomes even more pronounced for the cases with larger p_{m} values, further amplifying the aggregate disruptive effect of both reproduction operators[32]. Again, the often suggested solution to this problem would be hybridization, that is, combining the present algorithm with a better local search method such as SA or any type of a neighborhood search procedure. Note also that the average quality of the best individuals in the initial populations, and hence, also the quality of the starting point of the optimization, is generally slightly higher for larger populations, as the probability of generating a fitter individual increases with p.

The quality and efficiency of a particular GA parameter setting should not be measured only in terms of the best fitness value found, denoted by $k^*_{\text{eff}} \equiv k^*_{\text{eff}}(200)$ here, or the corresponding success rate. Equally important, especially for the loading-pattern optimization, is also the required computing time, usually expressed by the number of the objective function evaluations[33]. In this respect, all GAs with the elitist replacement strategy convincingly outperform the GAs with the pure generational replacement strategy (cf. Table 4.3). For instance, only around 20 % of the number of evaluations is needed for the GAs with

[32]It should be mentioned that the `RepairOperator()` function introduces an additional disruptive element into the GA.

[33]Remember that some individuals may have already been evaluated over the course of the optimization and, therefore, do not need to be (re-)evaluated again (see page 108).

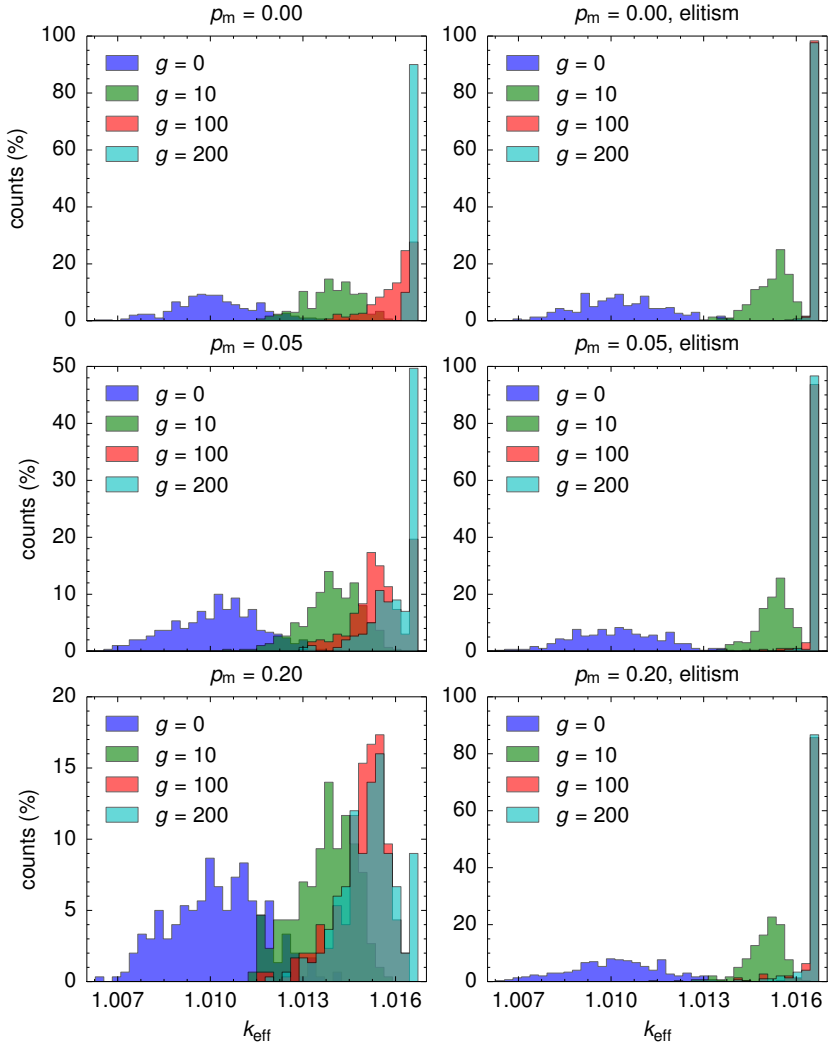


Figure 4.9: Problem 1 results (GA): evolution of the distribution of the fitness values (k_{eff}) in the population of $p = 30$ individuals, for different p_{m} values, and with or without elitism; the values were obtained by processing results of 10 independent runs and sorting them into 40 bins according to their fitness values

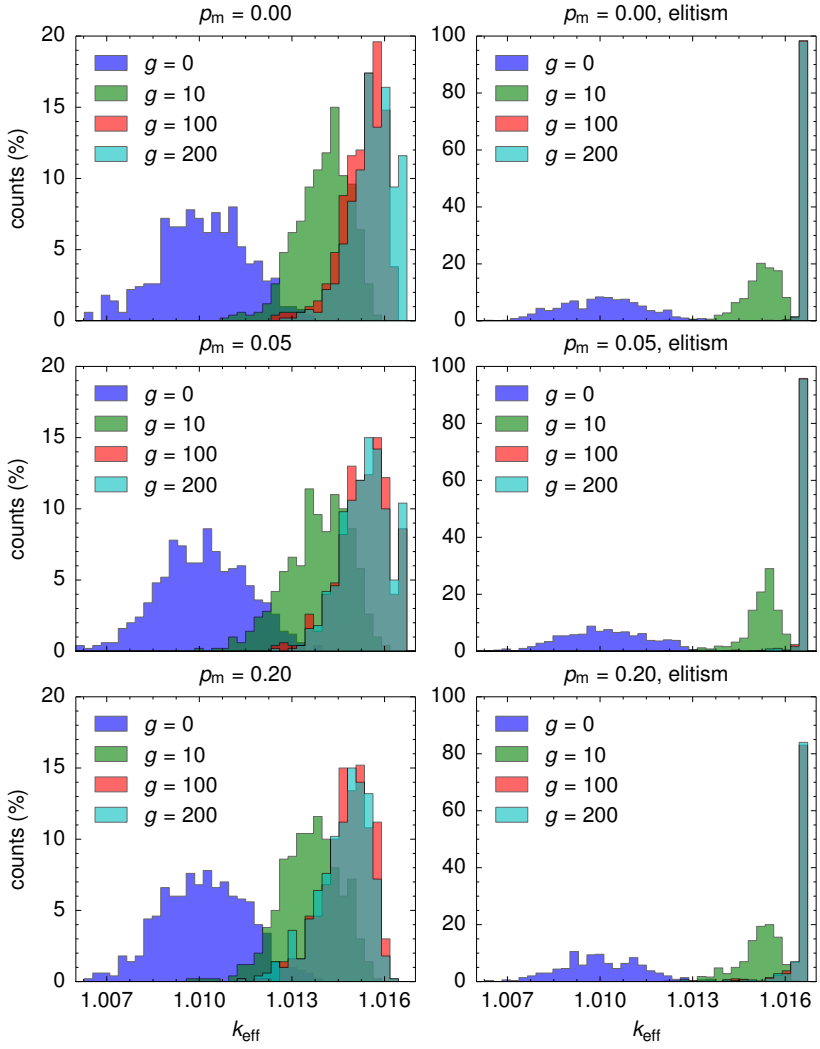


Figure 4.10: Problem 1 results (GA): evolution of the distribution of the fitness values (k_{eff}) in the population of $p = 50$ individuals, for different p_{m} values, and with or without elitism; the values were obtained by processing results of 10 independent runs and sorting them into 40 bins according to their fitness values

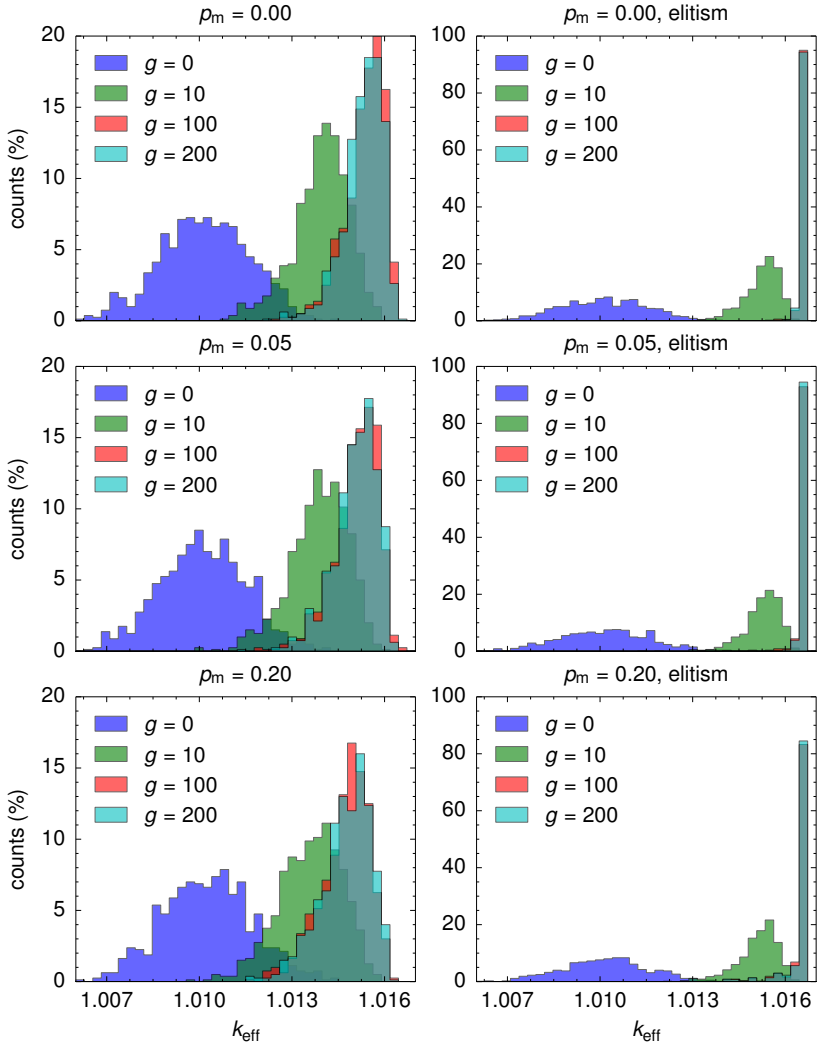


Figure 4.11: Problem 1 results (GA): evolution of the distribution of the fitness values (k_{eff}) in the population of $p = 80$ individuals, for different p_{m} values, and with or without elitism; the values were obtained by processing results of 10 independent runs and sorting them into 40 bins according to their fitness values

$p = 30$ when elitism is employed, the corresponding numbers for $p = 50$ and $p = 80$ being $\approx 15\,\%$ and $\approx 13\,\%$, respectively. As noted earlier in Section 4.3.1, the lower numbers of evaluations (unique solutions) mean also a lower diversity in a population and an increased risk of premature convergence. Knowing that the maximum number of possible evaluations equals $p \times (G + 1)$, it is apparent from Table 4.3 that almost all solutions (more than $90\,\%$) were unique when elitism was not used, with the exception of the cases with $p = 30$ and low p_m values 0.00 and 0.05, for which only $30\,\%$, respectively, $76\,\%$ of all solutions were unique. With elitism this fraction dropped below $20\,\%$ in all cases, meaning that even for the least diverse search without elitism more unique solutions were evaluated than for any of the searches with elitism ($3121 > 2300$, cf. Table 4.3).

It has been found that $k_\mathrm{eff}(\boldsymbol{x}) \in [1.00381, 1.01660]$ for any solution $\boldsymbol{x} \in \mathcal{F}$. In this respect, the largest attainable improvement in the fitness value equals $\Delta k_\mathrm{eff} = 1279\,\mathrm{pcm}$ and the most prevalent improvement achieved during the optimization, which starts from the most frequent average random value $k^*_\mathrm{eff}(0) \approx 1.01000$, equals $\Delta k_\mathrm{eff} \approx 660\,\mathrm{pcm}$. The corresponding gains in the operation times would then be $\approx 77\,\mathrm{EFPD}$ and $\approx 40\,\mathrm{EFPD}$, respectively, assuming that $\Delta k_\mathrm{eff} \approx 1500\,\mathrm{pcm}$ is needed for 90 EFPD of operation (cf. Table 3.9 and Table 3.10). Comparing to the originally proposed loading pattern from Figure 3.10, however, this improvement would be much lower, $\Delta k_\mathrm{eff} \approx 24\,\mathrm{pcm}$, which corresponds to additional $\approx 1.4\,\mathrm{EFPD}$ of operation only.

Overall, the GA characterized by $p = 80$, $p_\mathrm{m} = 0.05$, and elitist generational replacement strategy is suggested as a good GA to be used for solving Problem 2. It provides the highest consistency and success rate in finding the optimum solution among all investigated parameter settings, while holding the average number of needed full fitness-function evaluations reasonably low and still guaranteeing enough diversity in the search. A single optimization run would take approximately 17 minutes in this case, using the reference computing setup described at the beginning of Section 4.4.1.

ACO Results

The results of Problem 1 solved by the ACO algorithm are graphically presented in Figure 4.12 and summarized in Table 4.4. 12 different parameter settings were used to solve the problem, consisting of all combinations of different ant population sizes $k \in \{30, 50, 80\}$, pheromone evaporation rates $\rho \in \{0.1, 0.2\}$, and STR parameter values $q_0 \in \{0.8, 0.9\}$ (see Table 4.2). The value of the parameter γ from the global updating rule (4.7), was set to $\gamma = 1.992$, so that the product $\rho\,\gamma\,k_\mathrm{eff}(\boldsymbol{x})$ from the rule (4.7) is at least twice as large as the

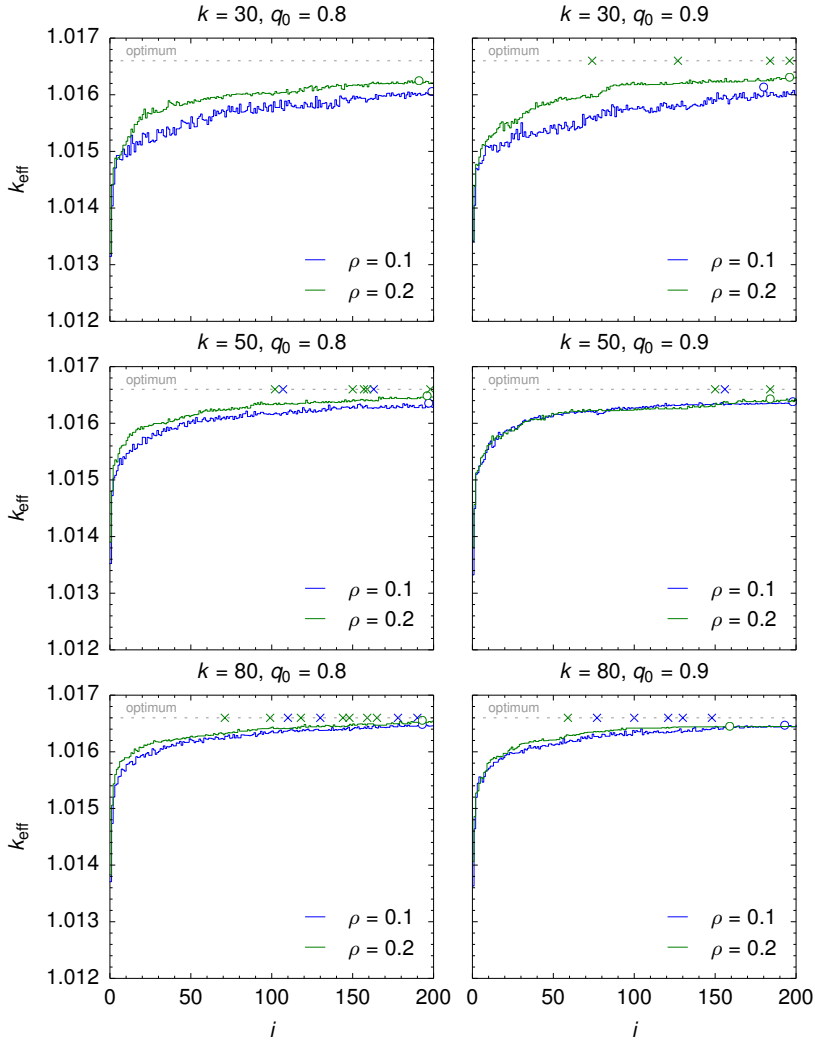


Figure 4.12: Problem 1 results (ACO): evolution of the iteration-best objective-function value (k_{eff}) obtained for different k, ρ, and q_0 values; the values were obtained by averaging results of 10 independent runs; "×" indicates the first time (if) the optimum solution was found for each of the 10 runs; "○" indicates the first time the maximum average value was achieved

Table 4.4: Problem 1 results (ACO)

#	ACO params.			k^*_{eff}		Optimum identification				ANE[a]
	k	ρ	q_0	Avg.	Max.	Count (#/10)	Avg. i	Min. i	Max. i	
1	30	0.1	0.8	1.01625	1.01658	0	–	–	–	5056
2	30	0.1	0.9	1.01626	1.01648	0	–	–	–	3229
3	30	0.2	0.8	1.01636	1.01654	0	–	–	–	5020
4	30	0.2	0.9	1.01642	1.01660	4	145	74	196	3236
5	50	0.1	0.8	1.01649	1.01660	2	135	107	163	8206
6	50	0.1	0.9	1.01643	1.01660	1	156	156	156	5042
7	50	0.2	0.8	1.01656	1.01660	2	135	107	163	8145
8	50	0.2	0.9	1.01651	1.01660	2	167	150	184	5068
9	80	0.1	0.8	1.01654	1.01660	4	152	110	190	12739
10	80	0.1	0.9	1.01655	1.01660	5	115	77	148	7781
11	80	0.2	0.8	1.01659	1.01660	7	129	71	165	12740
12	80	0.2	0.9	1.01646	1.01660	1	59	59	59	7750

[a] Average number of evaluations.
The values in each row were obtained by processing results of 10 independent runs.

product $\rho\,\tau_0$ from the local updating rule (4.6) for every feasible solution $\boldsymbol{x} \in \mathcal{F}$ (see the lowest k_{eff} value indicated in Figure 4.8).

In general, better results were obtained for the cases with larger ant populations and the lower of the two q_0 values. Compared to these two parameters, the influence of the evaporation rate ρ on the quality of the results was of a secondary importance. This can be explained by the role each of these parameters has in the algorithm. While the first two parameters affect the diversity of the search, the latter one controls the amplitude and the speed of the dynamic feedback effect, with which the algorithm responses to the changing quality of the solutions during optimization Indeed, the more ants in the population and the larger the chance of the roulette-wheel method to choose the next ant move, the more likely a new unique solution to be constructed of a potentially higher quality. And also, the larger the pheromone evaporation rate, the more pheromone to be deposited on the best-solution path and the more pheromone to be evaporated from all other paths in fewer iterations, at the same time.

Indeed, it follows from Figure 4.12 that the iteration-best k_{eff} values averaged over 10 runs with a different seed show the steepest increase at the beginning of optimization for the cases with $k = 80$ ants and the slowest increase for the cases with $k = 30$ ants. Also, the cases with the larger ρ value seem to converge faster and get closer to the optimum value, as the higher-quality solutions

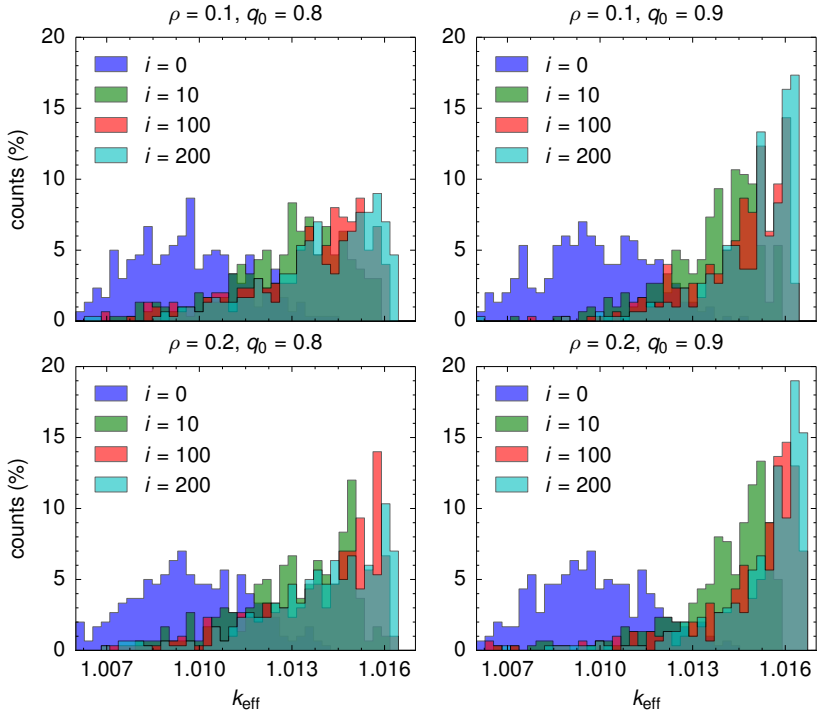


Figure 4.13: Problem 1 results (ACO): evolution of the distribution of the objective-function values (k_{eff}) in the iteration of $k = 30$ ants, for different ρ and q_0 values; the values were obtained by processing results of 10 independent runs and sorting them into 40 bins according to their objective-function values

have a better chance to be promoted, which, on the other hand, increases the probability of a premature convergence, especially for small ant populations. Other evidence that demonstrates the relation between the q_0 values and the diversity of the search are the numbers of evaluations indicated in the last column of Table 4.4. It is apparent from the numbers that around 80 % of the maximum number of $k \times (I + 1)$ evaluations was performed when q_0 was equal to 0.8, which is approximately 30 % more than when q_0 was 0.9. These numbers are representative for all combinations of both ρ values and all three ant population sizes.

The impact the different k, q_0, and ρ values have on the distribution of the objective-function values throughout the optimization is further illustrated in Figure 4.13–4.15. In correspondence with what was just said, it can be seen

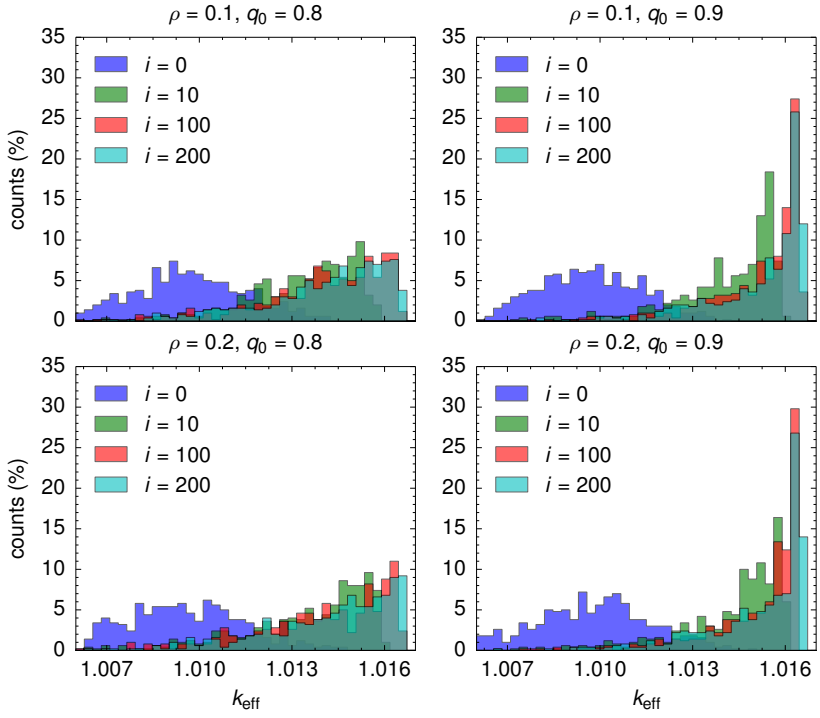


Figure 4.14: Problem 1 results (ACO): evolution of the distribution of the objective-function values (k_{eff}) in the iteration of $k = 50$ ants, for different ρ and q_0 values; the values were obtained by processing results of 10 independent runs and sorting them into 40 bins according to their objective-function values

from the figures that the value of q_0 influences the algorithm's efficiency more than that of ρ and, also, that larger ρ values become more favorable with the growing k number.

The known optimum value of the BOC effective multiplication factor k_{eff} was found seven out of 10 times after evaluating almost 13 000 trial solutions on average for $k = 80$, $q_0 = 0.8$, and $\rho = 0.2$. A slightly lower success rate was obtained for the case with $k = 80$, $q_0 = 0.9$, and $\rho = 0.1$, when the optimum was found five out of 10 times after approximately 8000 evaluations. As expected, the worst results were received for the smallest ant population. In fact, the optimum was not found a single time for three out of four ACO configurations with 30 ants. The only exception was the configuration with $q_0 = 0.9$ and $\rho = 0.2$, which delivered the optimum solutions four times out of

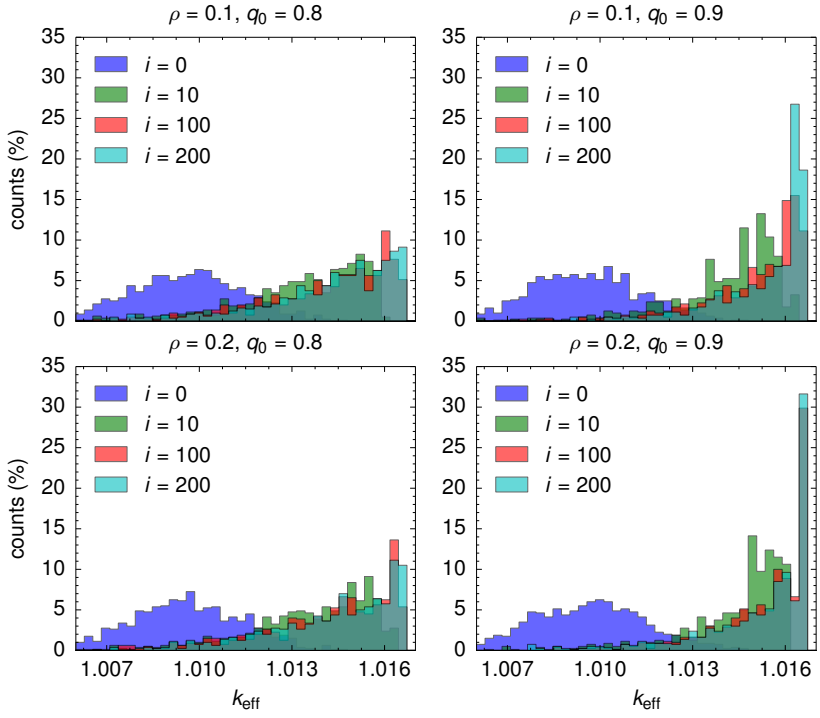


Figure 4.15: Problem 1 results (ACO): evolution of the distribution of the objective-function values (k_{eff}) in the iteration of $k = 80$ ants, for different ρ and q_0 values; the values were obtained by processing results of 10 independent runs and sorting them into 40 bins according to their objective-function values

10 after approximately 3000 evaluations and, as such, even outperformed all ACO configurations with 50 ants and one with 80 ants (cf. Table 4.4).

Figure 4.16 shows graphically how the distribution of the pheromone values changes in the search graph during optimization. The different segments in the figure represent different parts of those paths taken by the ants that correspond to different iteration-best solutions. The thickness of each line is used as a measure of the pheromone amount $\tau_{r\,s\,p}$ deposited between two connected nodes r and s at a position p and is proportional to $\tau_{r\,s\,p}$ subtracted by τ_0[34]. The thicker the line, the more the pheromone is deposited on it and the higher the chance that the ant will walk this way. It is clear from the figure, what was the vector representation of the first iteration-best solution (see the top left plot

[34]Hence, there would be no lines in the plot for $i = 0$, since $\tau_{r\,s\,p} = \tau_0$ for all $\tau_{r\,s\,p} \in \mathcal{T}$.

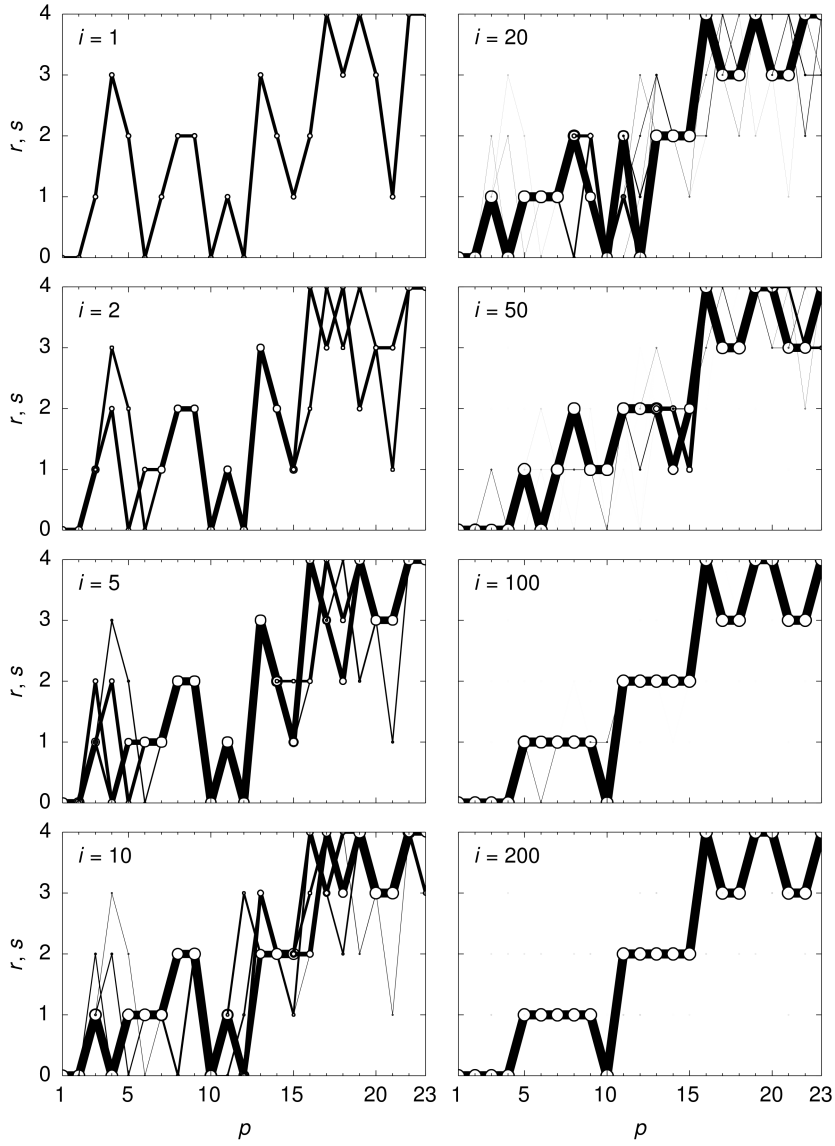


Figure 4.16: Problem 1 results (ACO): evolution of the pheromone distribution for $k = 30$ ants, $\rho = 0.2$, and $q_0 = 0.9$ (the optimum was first identified at iteration $i = 74$); the line thicknesses correspond to $\tau_{r\,s\,p} - \tau_0$; the maximum pheromone value at iteration $i = 200$ was $\tau = 1.569$

for $i = 1$): $\boldsymbol{x} = (0, 0, 1, 3, 2, 0, 1, 2, 2, 0, 1, 0, 3, 2, 1, 2, 4, 3, 4, 3, 1, 4, 4)$. The second iteration-best solution can also be deduced from the figure, as well as the third one, forth one, and a few more. It can also be seen from the figure, how the pheromone amount increases or decreases between two nodes, depending on whether the ant delivering the iteration-best solution takes the same direction at a position p or not. For instance, $\tau_{0\,1\,2}$ increased whereas $\tau_{1\,3\,3}$ decreased between the iteration 1 and 2. Another interesting observation based on the analysis of Figure 4.16 is that even if some path is strongly preferred (marked with a lot of pheromone) at a certain iteration, this does not automatically imply a premature convergence and can change in subsequent iterations, as follows from the comparison of the two plots for $i = 50$ and $i = 100$, for instance. The concrete values depicted in Figure 4.16 were derived from the results of one of the optimization runs characterized by $k = 30$, $\rho = 0.2$, and $q_0 = 0.9$. The optimum solution was first found at iteration $i = 74$ and the maximum pheromone value at the last iteration was $\tau = 1.569$ in this case.

Also interesting is the comparison with the results obtained by applying the GA. As follows from Table 4.3 and Table 4.4, the opted GA implementation clearly outperformed the one of the ACO, only if one considers the use of the elitist population replacement strategy in the GA[35]. In this case, the optimum solution can be found both faster and with a larger success rate using the GA than the ACO algorithm. Moreover, significantly fewer solution evaluations are performed when the elitist GA is used, which makes it even more the preferred method for the solution of the current problem. However, the dominance of the GA becomes less obvious if one compares only the results obtained without employing the elitism. In that case, the performances of both algorithms become comparable in terms of the success rate of finding the optimum solution. However, since better results are obtained for the GA with the smallest population of 30 individuals and for the ACO with the largest population of 80 ants, the GA still remains the preferred method. One of the suggested ways how to improve the ACO's efficiency and make it a more competitive optimization algorithm, is to use the best-so-far solutions for the pheromone update (i.e., elitism) and/or to apply a local search method to the constructed solutions. Both algorithms exhibit generally different convergence characteristics. One of the problems encountered with the GA was that the algorithm begins to stagnate after a certain number of iterations, due to the disruptive effect of the reproduction operators (see the preceding part on the GA results). However, this is not the case of the ACO algorithm. It can be seen from Figure 4.12 that all curves keep rising steadily through all 201 iterations for all combinations of the parameter values. Moreover, the range of the objective-function values characteristic for a population of solutions at an advanced stage of optimization is much broader

[35] Some form of elitism can also be implemented in ACO, as was commented on in Footnote 27 in Section 4.3.2. However, this option was not considered in the present work.

for the ACO than for the GA, meaning a larger diversity in the population and, hence, also a higher resistance against a premature convergence (cf. the different histograms in Figure 4.9–4.11 and Figure 4.13–4.15). Therefore, one may consider the ACO algorithm a preferred optimization method, when similar but strongly constrained problems have to be solved.

The ACO configuration characterized by $k = 30$, $q_0 = 0.9$, and $\rho = 0.2$ will be used to solve Problem 2, despite the fact that the configurations with 80 ants led to generally better results in 201 iterations. This choice was made assuming the same maximum permitted number of evaluations as when the GA was applied to the problem (8080). Therefore, a lower number of ants was chosen that allows for a larger number of iterations rather than a large number of ants and a limited number of iterations.

4.4.2 Problem 2

As outlined in Section 4.1 at the beginning of this chapter, the second problem is proposed as an example of a loading pattern optimization problem that could prospectively be considered as a realistic MYRRHA-FASTEF in-core fuel management problem to be solved once the facility will become operational. It aims at maximizing the average fast-neutron fluence at the IPS channels over a single fuel cycle, given the identical fuel inventory constraints like in Problem 1, while imposing additional constraints on the maximum cladding temperature and EOC core criticality.

Problem Definition

Let Problem 2 be defined as follows:

$$\underset{\boldsymbol{x}}{\text{Maximize}} \quad \bar{\Theta}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, P) \ , \tag{4.10}$$

$$\text{subject to} \quad \sum_{i=1}^{N_{\text{FA}}} \delta_{x_i\, t_j} \leq m_j, \qquad j = 1, 2, \ldots, N \ , \tag{4.11}$$

$$T_{\text{clad}}(\boldsymbol{x}, P, t) \leq T^{\text{lim}}_{\text{clad}} \ , \tag{4.12}$$

$$k^{\text{EOC}}_{\text{eff}}(\boldsymbol{x}, P) \geq 1 \ , \tag{4.13}$$

$$0 < P \leq P_{\text{lim}} \ , \tag{4.14}$$

$$0 \leq t \leq T_c \ , \tag{4.15}$$

where $\bar{\Theta}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, P)$ is the average fast-neutron fluence in the IPS channels characteristic for the loading pattern $\boldsymbol{x}$ and the reactor power P, which should be lower than or equal to the limit value $P_{\text{lim}} = 100\,\text{MW}$; $T_{\text{clad}}(\boldsymbol{x}, P, t)$ is the maximum cladding temperature achieved at any time t during the fuel cycle of the length $t_{\text{c}} = 90\,\text{EFPD}$; $k^{\text{EOC}}_{\text{eff}}(\boldsymbol{x}, P)$ is the effective multiplication factor at the EOC ($t = T_{\text{c}}$); $T^{\text{lim}}_{\text{clad}} = 550\,^{\circ}\text{C}$ is the maximum cladding-temperature limit[36], and the rest of the notation and parameter values is the same as in Problem 1.

The neutron fluence $\Theta(\boldsymbol{r}, E)$ is defined here as the neutron flux $\phi(\boldsymbol{r}, E, t)$ integrated over the time interval $[t_1, t_2]$:

$$\Theta(\boldsymbol{r}, E) = \int_{t_1}^{t_2} \phi(\boldsymbol{r}, E, t)\ \mathrm{d}t\ . \tag{4.16}$$

Hence, for $\bar{\Theta}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, P)$ from Eq. (4.10), assuming $t_1 = 0$, and $t_2 = T_c$:

$$\bar{\Theta}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, P) = \frac{1}{V_{\text{IPS}}} \int_{E_1}^{E_2} \int_{V_{\text{IPS}}} \Theta(\boldsymbol{x}, \boldsymbol{r}, E, P)\ \mathrm{d}^3 r\ \mathrm{d}E\ , \tag{4.17}$$

where E_1 and E_2 are the boundaries defining the interval of fast neutron energies and the spatial vector goes through the IPS volume in the core fissile zone V_{IPS}.

In practice, however, $\Theta(\boldsymbol{r}, E)$ is calculated as if the IPS volume was irradiated by a constant fast-neutron flux equal to the average of the BOC and EOC flux values. This is possible, since the average fast-neutron flux in the IPS region $\bar{\phi}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, t)$ can be very well fit by a linear function of time; see Figure 4.17, in which $\bar{\phi}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, t)$ is plotted for the two upper loading patterns from Figure 4.19. Thus, using the multi-group nodal formalism:

$$\bar{\Theta}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, P) = \frac{1}{2}\, T_{\text{c}} \left(\bar{\phi}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, P, 0) + \bar{\phi}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, P, T_{\text{c}}) \right)\ , \tag{4.18}$$

where

$$\bar{\phi}^{\text{fast}}_{\text{IPS}}(\boldsymbol{x}, P, t) = \frac{1}{s} \sum_{g=1}^{4} \sum_{i=1}^{k} s_i\, \phi_{g\,i}(\boldsymbol{x}, P, t)\ , \tag{4.19}$$

the four energy groups $g = 1$ to $g = 4$ cover the neutron energies from 0.5 MeV to 14.2 MeV in the adopted 15-group scheme; s_i are the IPS cell frequencies corresponding to the k IPS nodes in the core fissile zone (see Figure 3.4 and 4.5), and $s = \sum_{i=1}^{k} s_i$.

[36]The larger of the two fuel-cladding temperature limits mentioned in Section 2.4.4 is used here, as the MYRRHA-FASTEF reactor core and fuel were originally designed with this particular limit in mind. Also, the fuel compositions corresponding to the different FA types considered in Problem 2 were obtained based on an equilibrium cycle analysis for $P = 100\,\text{MW}$.

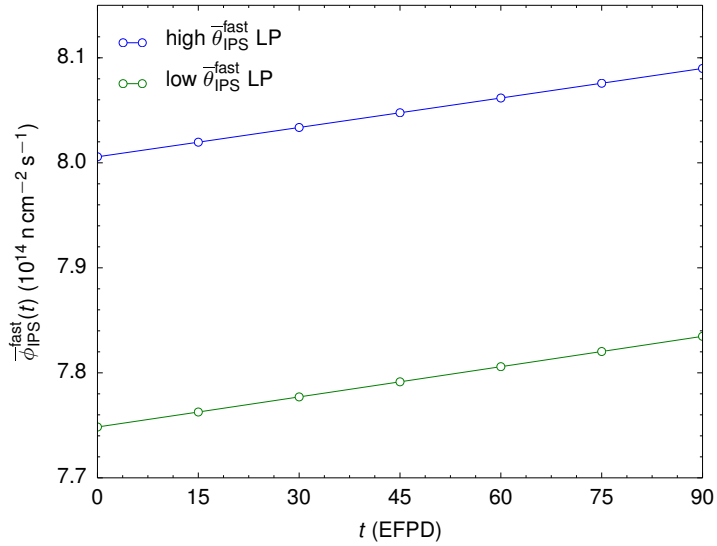


Figure 4.17: Evolution of the IPS-averaged fast-neutron flux

Solution Methods

The GA and ACO methods from Problem 1 were used for solving Problem 2. The parameters of both algorithms were set to the values obtained based on the analysis of the Problem 1 results, as discussed at the end of the previous section. The only exception is the maximum number of generations that was decreased to $G = 100$ for the GA and increased to $I = 268$ for the ACO. Also, all calculations were performed using the same MYRRHA-FASTEF neutronics model and computing platform as in Problem 1 (see Section 4.4.1).

Compared to Problem 1, additional constraints are added in Problem 2, besides the fuel inventory constraints that are common for both problems and used in the adopted solution scheme to form candidate solutions. While the fuel inventory constraints are satisfied "a priori", that is, before the solutions are evaluated, the compliance with the other constraints can be verified only "a posteriori", that is, after the solutions are evaluated. Therefore, some constraint-handling strategy has to be applied to the problem to deal with the latter type of constraints. At this point, before this strategy will be discussed in the following part, it is worthwhile to make a remark on the calculated $k_{\text{eff}}^{\text{EOC}}$ values.

EOC criticality correction. The EOC effective multiplication factor $k_{\text{eff}}^{\text{EOC}}(\boldsymbol{x}, P)$ is one of the loading-pattern characteristics calculated for each candidate solution in Problem 2, see Eq. (4.13). Its estimation is usually very sensitive to the employed neutronics model and nuclear data and, as such, it may result in quite different values. In this respect, the diffusion REBUS model, developed in Chapter 3 and used in Section 3.3.2 for evaluating the MYRRHA-FASTEF BOEC core, gives satisfactory results, comparing to the reference MCNPX results. As follows from Table 3.9 and Table 3.10, the REBUS model overestimates k_{eff} by 147 pcm at BOC and by 56 pcm at EOC. Therefore, all $k_{\text{eff}}^{\text{EOC}}(\boldsymbol{x}, P)$ values calculated hereinafter in this section will be automatically decreased by 100 pcm,

$$k_{\text{eff}}^{\text{EOC}}(\boldsymbol{x}, P) \longleftarrow k_{\text{eff}}^{\text{EOC}}(\boldsymbol{x}, P) - 100\,\text{pcm} \;, \tag{4.20}$$

in order to mitigate this discrepancy.

Constraints handling

Two additional constraints are introduced in Problem 2: the maximum fuel-cladding temperature constraint, defined by Eq. (4.12) and (4.15), and the EOC criticality constraint, defined by Eq. (4.13). Both constraints depend on the reactor power P, which acts in the current formulation of the problem as a free variable of a value from the range given by Eq. (4.14). It should be noted that the first-named constraint can have a simplified form

$$T_{\text{clad}}(\boldsymbol{x}, P, 0) \leq T_{\text{clad}}^{\text{lim}} \;, \tag{4.21}$$

as the largest power-peaking factors, and hence also the largest $T_{\text{clad}}(\boldsymbol{x}, P, t)$ values, are achieved at the BOC conditions ($t = 0$) for every critical MYRRHA loading pattern.

Using the opted solution scheme, with the loading pattern $\boldsymbol{x}$ as a single decision variable evaluated by a black-box reactor-analysis code, one can deal with the constraints (4.13) and (4.21) in two different ways, depending on the treatment of the variable P. In the first one, a candidate solution $\boldsymbol{x}$ is first evaluated applying the simplified analysis scheme from Figure 3.1 for an arbitrary user-specified value of P, and the augmented objective function $f_{\text{a}}(\boldsymbol{x})$ is then calculated, using the standard penalization technique described later in this section. In scheme 3.1, the neutron flux $\phi(\boldsymbol{r}, E, t)$ from Eq. (4.16), and thence also the objective function $\bar{\Theta}_{\text{IPS}}^{\text{fast}}(\boldsymbol{x}, P)$ from Eq. (4.10), is obtained as a nontrivial solution of the eigenvalue problem (2.21) normalized according to Eq. (2.22) to the given value of $P > 0\,\text{MW}$. All other loading-pattern characteristics are thereby also to be calculated for the same value of P. In this case, however, the number of considered candidate solutions and thus also the quality of the optimum

solution depend on the particular preset value of P, which, for facilities like MYRRHA, should be perceived rather as one of the decision variables than a fixed input parameter. Moreover, if P is set too large (e.g., $P = P_{\mathrm{lim}}$), which further reduces the search space size, the cladding temperature limit becomes difficult to satisfy. Consequently, the feasible solutions become hard if not impossible to find. Therefore, knowing that $T_{\mathrm{clad}}(\boldsymbol{x}, P, 0)$ from the constraint (4.21) is calculated as a linear function of P[37], while holding the other arguments fixed, the following three-step approach is proposed for the objective-function evaluation, as one of the practical ways to deal with this difficulty[38]: In the first step, the loading-pattern $\boldsymbol{x}$ is evaluated for the BOC conditions and an arbitrary value of P denoted by P'; then, in the second step, $\boldsymbol{x}$ is evaluated again, this time applying the simplified scheme 3.1 for the reactor power P derived as

$$P = \begin{cases} P' \frac{T^{\mathrm{lim}}_{\mathrm{clad}} - T_{\mathrm{in}}}{T_{\mathrm{clad}}(\boldsymbol{x}, P', 0) - T_{\mathrm{in}}} & \text{if } T_{\mathrm{clad}}(\boldsymbol{x}, P', 0) > T^{\mathrm{lim}}_{\mathrm{clad}}, \\ P' & \text{otherwise,} \end{cases} \tag{4.22}$$

where $T_{\mathrm{in}} = 200\,^{\circ}\mathrm{C}$ is the coolant inlet temperature; and, finally, $f_{\mathrm{a}}(\boldsymbol{x})$ is calculated in the last step. In this way, one arrives at a solution, for which the constraint (4.21) is always implicitly satisfied, leaving Eq. (4.13) the only acting constraint in Problem 2[39]. In loose terms, the approach just described may be understood as one relying on $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}(\boldsymbol{x})$ "normalization" to $T^{\mathrm{lim}}_{\mathrm{clad}}$, with $T_{\mathrm{clad}}(\boldsymbol{x}, P, t)$ being fixed and P being an output variable. This should be compared to the previous more-common approach based on $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}(\boldsymbol{x}, P)$ normalization to an input value of P, with $T_{\mathrm{clad}}(\boldsymbol{x}, P, t)$ being an output variable (cf. the two different schemes in Figure 4.18).

Figure 4.19 demonstrates the impact of applying both approaches to examples of four loading patterns, two of them being characterized by high and two of them by low $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}(\boldsymbol{x})$ values. It is evident from the figure that when the first approach is applied with the reactor power fixed, the neutron fluence in the inner-core region, where the IPSs are located, is increased or decreased primarily by changing the peaking factor P_{q}. In case of the second approach with the constant maximum cladding temperature, this can also be done by increasing, respectively, decreasing the reactor power, adding more variability to the optimization. Neutron leakage from the core region is an important factor for both approaches. Comparing the corresponding values from Figure 4.19[40], one

[37] See the description of the T-H model in Appendix B.

[38] An alternative approach would be to include P into the vector representation of the solution as one of the decision variables, and reformulate the original discrete problem (4.10) as a mixed-integer problem. $\max_{\boldsymbol{x}} \bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}(\boldsymbol{x}, P) \rightarrow \max_{\boldsymbol{x}, P} \bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}(\boldsymbol{x}, P)$.

[39] It is assumed that the inequalities (4.11) are also satisfied, since only those solutions that comply with the fuel-inventory constraints are evaluated.

[40] The characteristics that correspond to the upper-left LP from Figure 4.19 for $T_{\mathrm{clad}} = 549.9\,^{\circ}\mathrm{C}$ are: $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}} = 6.115\,\mathrm{n\,kb^{-1}}$, $k^{\mathrm{EOC}}_{\mathrm{eff}} = 0.99974$, $P = 97.7\,\mathrm{MW}$, and $P_{\mathrm{q}} = 1.719$.

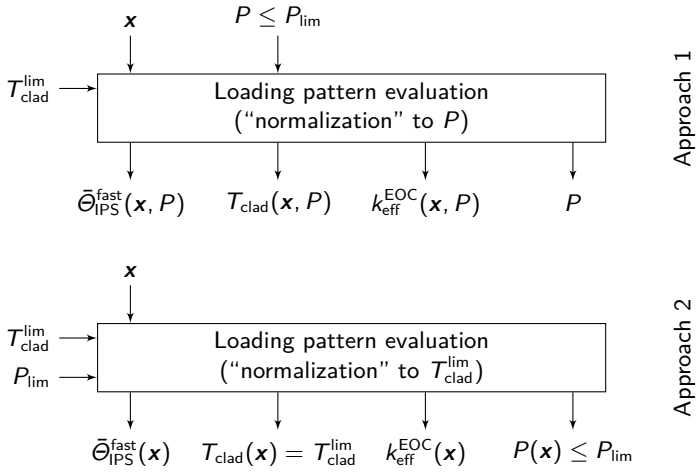


Figure 4.18: Two loading-pattern evaluation approaches

can easily observe that better results were obtained when the second approach was used to maximize or minimize $\bar{\Theta}_{\text{IPS}}^{\text{fast}}$, This corresponds with the expectations.

With the fuel-inventory and maximum cladding temperature constraints being treated appropriately in a way described above, the last remaining constraint to be handled is the EOC criticality condition (4.13). Since the EAs are unconstrained optimization procedures (i.e., procedures that do not treat constraints explicitly), it is necessary to find ways of incorporating the constraints into the fitness function. A comprehensive review of the most important constraint-handling techniques developed for EAs was given by Coello Coello (2002). This review includes a wide variety of techniques that go from several variations of a simple penalty function, through special representations, genetic operators, and repair algorithms[41], to more advanced techniques that, for instance, treat objectives and constraints separately (e.g., co-evolution, multi-objective optimization, dominance relation based techniques, behavioral memory, or superiority of feasible points techniques), or emulate the behavior of the immune system, culture, and others. In the present work, the objective-function penalization technique was used as a simple and effective way to deal with the single EOC criticality constraint.

[41] These techniques have already been discussed in detail in Section 4.3.1. Note also that both methods used to deal with the fuel inventory and maximum cladding temperature constraints fall into the category of repair algorithms.

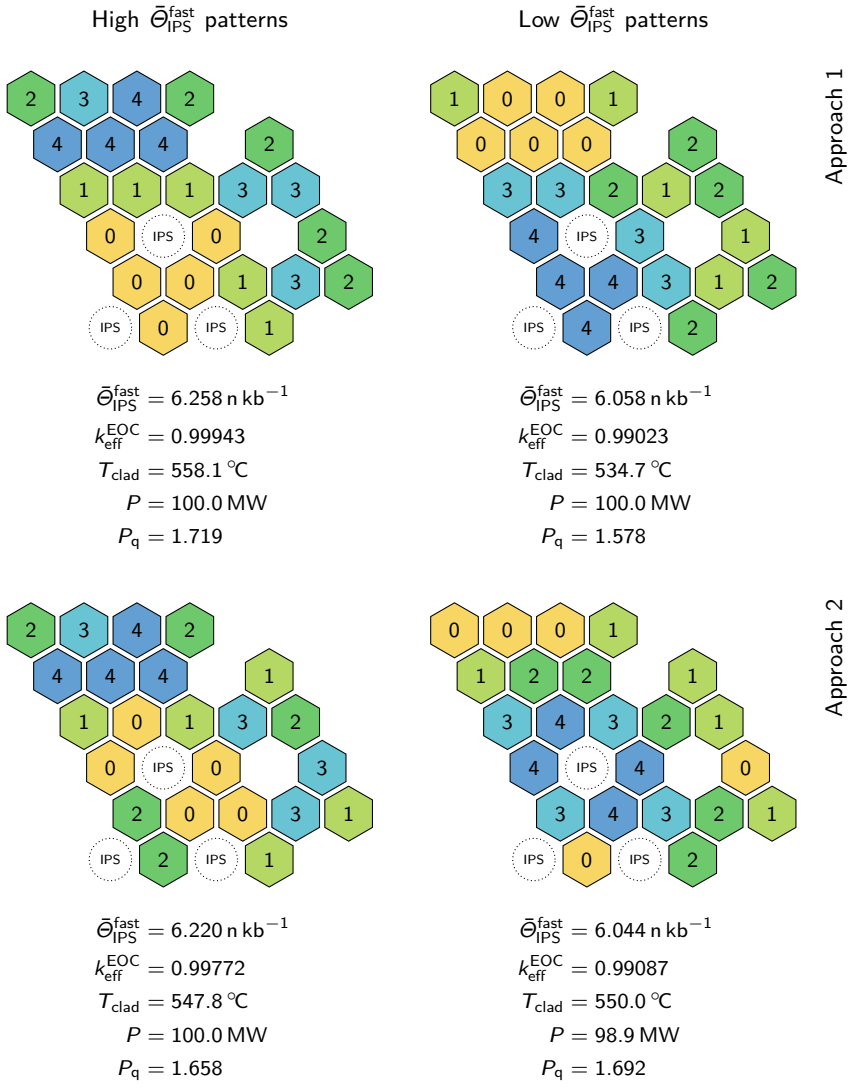


Figure 4.19: High- and low-IPS-fluence loading patterns obtained by applying two loading-pattern evaluation approaches

Objective function penalization. Penalty functions are the oldest and the most common approach used to incorporate constraints and particularly inequality constraints into unconstrained optimization algorithms such as the EAs. The general idea behind the penalty functions is to transform a constrained optimization problem into an unconstrained problem by adding (or subtracting) a certaing value to (from) the objective function, based on the constraint violation present in a certain solution (Coello Coello, 2002). Penalty functions are very easy to implement, computationally inexpensive, and can also be very efficient. On the other hand, there is no general guideline on designing them and constructing an efficient penalty function is usually highly dependent on the problem at hand (Gen and Cheng, 1996). Also, although the number of parameters is quite limited, the chosen parametrization has still a big influence on the technique's effectivity and efficiency, which will be documented later in the text.

The most common addition form of the penalty function was outlined in Section 2.2.4 in Eq. (2.13). However, some other forms of the penalty function exist, such as a multiplication form or other forms, mainly depending on the properties of the involved parameters (e.g., Gen and Cheng, 1996): penalties with or without parameters, constant versus variable penalties, etc. Coello Coello (2002) distinguishes between exterior and interior penalty functions that either start from an infeasible solution and move toward the feasible region (exterior functions) or start from a feasible region and stay there, since the constraint boundaries act as barriers during the optimization process (interior functions). The exterior penalty functions are usually considered for EAs, because they do not require an initial feasible solution. Finding such an initial solution may already be a very difficult problem itself.

Three types of penalization were considered for solving Problem 2, each of them characterized by a different form of a penalty factor ω from Eq. (2.13)[42]. The first type relies on the static penalty that does not depend on the current generation number in any way and remains constant during the entire evolutionary process. In this case, the optimized augmented function is of the form

$$f_{\mathrm{a}}(\boldsymbol{x}) = \bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}(\boldsymbol{x}) + \omega \left(1 - k^{\mathrm{EOC}}_{\mathrm{efff}}(\boldsymbol{x})\right)^{\alpha} \ , \tag{4.23}$$

where[43]

$$\omega \begin{cases} = 0 & \text{if } k^{\mathrm{EOC}}_{\mathrm{eff}}(\boldsymbol{x}) > 1, \\ < 0 & \text{otherwise,} \end{cases} \tag{4.24}$$

[42] $m = 1$ in this case, $\omega_1 \equiv \omega$.

[43] The first part of the rule (4.24) applies to all forms of the penalty factors discussed in this text. In other words, the objective function $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}(\boldsymbol{x})$ is not penalized if the constraint (4.13) is satisfied.

and the term in the brackets is a measure of the degree of violation of the constraint[44]. A slightly modified version of the penalization factor tried in the optimization was the linear dynamic penalty

$$\omega(g) = g\,\frac{\omega_2 - \omega_1}{G} + \omega_1\ , \tag{4.25}$$

which gradually changes[45] from the value ω_1 to the value $\omega_2 \neq \omega_1$ throughout the whole optimization process of G iterations[46]. These first two penalizations are ignorant of the ongoing success (or lack thereof) of the search and, therefore, cannot guide the search to particularly attractive regions away from those already visited and unattractive ones. The third and the last penalty applied here tries to mitigate this by introducing an additional parameter k into the formula for ω. This is done in the following way:

$$\omega(g,k) \leftarrow \begin{cases} \beta_1\,\omega(g,k) & \text{if all the best solutions in the last } k \text{ iterations were infeasible,} \\ \beta_2^{-1}\,\omega(g,k) & \text{if all the best solutions in the last } k \text{ iterations were feasible,} \\ \omega(g,k) & \text{otherwise; that is, if some of the best solutions in the last } k \text{ iterations were feasible and some of them infeasible.} \end{cases} \tag{4.26}$$

In Eq. (4.26), $\beta_1 \neq \beta_2$ in order to avoid cycling[47].

Despite the relatively small number of parameters in Eq. (4.23)–(4.26), finding their right values still remains the most challenging aspect of applying the penalty-based constraint-handling technique[48]. Generally, when the factors are too lenient, the final solution may be infeasible, because most of the search time will be spent exploring the infeasible region. Whereas, if the factors are too severe, the EA may converge to a non-optimal feasible solution far from the boundary with the infeasible region, where global optima are often located. Moreover, a large penalty discourages the exploration of the infeasible region and, as such, has a tendency to move to one of the disjoint feasible regions (if they are disjoint) and stay there[49]. According to the minimum penalty rule

[44] In ideal case, the penalty amount should also consider the distance to optima, besides the distance from the feasible region. However, this is usually impossible due to the lack of any a priory knowledge about the optima (Gen and Cheng, 1996).

[45] Note that ω from Eq. (4.23) must be negative, $\omega \in \mathbf{R}^-$, so it can penalize an objective function that has to be maximized in this case.

[46] Note that the symbols I and i are used in the ACO instead of the symbols G and g used in the GA.

[47] Usually $\beta_1 > \beta_2$.

[48] For this reason, different ways to automate the definition of good penalty factors have been proposed recently by researchers in evolutionary computing (Coello Coello, 2002).

[49] An extreme case is the so called "death penalty", which automatically rejects all infeasible solutions during the search.

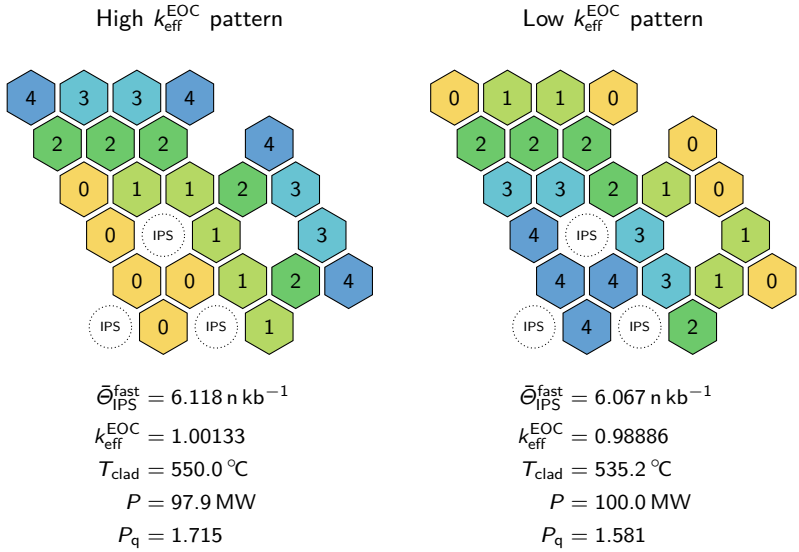


Figure 4.20: High- and low-$k_{\text{keff}}^{\text{EOC}}$ loading patterns

(Coello Coello, 2002), "the penalty should be kept as low as possible, just above the limit below which infeasible solutions are optimal."

In case of Problem 2, the values of the penalization parameters ω, ω_1, ω_2, β_1, β_2, k, and $\omega(0,k)$ were mostly chosen based on the trial-and-error method. The initial value of the static penalty factor ω was estimated as follows: first, the ranges of possible $\bar{\Theta}_{\text{IPS}}^{\text{fast}}$ and $k_{\text{eff}}^{\text{EOC}}$ values were identified (see Figure 4.19 and Figure 4.20); and then, ω was calculated as a scaling factor that makes a unit increase (decrease) of $\bar{\Theta}_{\text{IPS}}^{\text{fast}}$ within its boundary values equal to a unit decrease (increase) of $k_{\text{eff}}^{\text{EOC}}$ within the range given by its minimum value and the criticality criterion:

$$\omega = -\frac{6.220 \times 10^{21} - 6.044 \times 10^{21}}{1.00000 - 0.98886} \approx -1.58 \times 10^{22}\ . \tag{4.27}$$

From this value other ω-values used in optimization were deduced that will be specified later in the part on the discussion of the optimization results. Parameter values $\alpha = 0$, $\beta_1 = 1.05$, $\beta_2 = 1.03$, and $k = 25$ were used in all cases.

GA Results

Static penalty. The first optimization was done for the static penalty factor $\omega = -1.58 \times 10^{22}$. The corresponding results are shown in Figure 4.21 and Figure 4.22. The first figure shows the evolution of the population-best individuals in terms of the maximized fitness-function f_{a} values and the corresponding $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}$ and $k^{\mathrm{EOC}}_{\mathrm{eff}}$ values. "×" in Figure 4.21 indicates the first time the largest f_{a}, respectively, $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}$ value was found, and "○" indicates the $k^{\mathrm{EOC}}_{\mathrm{eff}}$ value associated with the individual with the largest $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}$ value. The second figure illustrates the evolution of the distribution of the objective-function $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}$ values through six generations $g = 0$, 5, 10, 25, 50, and 100. It can be seen from both figures that the GA did not converge to a feasible solution in this case, because the selection pressure toward individuals with larger $k^{\mathrm{EOC}}_{\mathrm{eff}}$ values was too small and solutions with larger $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}$ were preferred instead to increase the maximized f_{a} value. Therefore, additional optimizations were performed for increased ω values to obtain a high-quality feasible solution. It was observed that the algorithm started to deliver feasible solutions for penalties three-times larger than the previous value and higher. The results obtained for $\omega = -4.74 \times 10^{22}$ are presented in Figure 4.23 and Figure 4.24. Indeed, if one compares the distribution of the individuals in the last population ($g = 100$) for both ω values (cf. Figure 4.22 and Figure 4.24), it can be noticed that, in the first case, almost all individuals are accumulated higher in the $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}$ direction below the EOC criticality limit. In the second case, the majority of the individuals is located lower above the criticality limit. The influence of the ω value on the direction of the search is further demonstrated in Figure 4.25, where the progression of the movement of the population-best individual through the "f_{a}-landscape" is graphically illustrated for three different ω values. As expected, the evolution for the lowest ω starts from the point characterized by a relatively large $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}$ and relatively low $k^{\mathrm{EOC}}_{\mathrm{eff}}$ value, as the latter one plays a less important role in this case. The initial point of the optimization moves in the bottom-right direction as ω increases, and larger f_{a} values become to be achieved mainly due to the improvement in the $k^{\mathrm{EOC}}_{\mathrm{eff}}$ domain (basically no movement to the left can be observed in the bottom plot in Figure 4.25).

Another interesting observation based on the study of Figure 4.21 and Figure 4.23 is that individuals with large $\bar{\Theta}^{\mathrm{fast}}_{\mathrm{IPS}}$ values are already present in the initial random population ($g = 0$) and that these values do not improve too much during the optimization. In fact, the improvement in the quality of the solution is achieved mainly by increasing the $k^{\mathrm{EOC}}_{\mathrm{eff}}$ values. This confirms the above conclusions.

A closer look at the distribution of the individuals in the initial population in Figure 4.22 and Figure 4.24 reveals that the randomly generated individuals

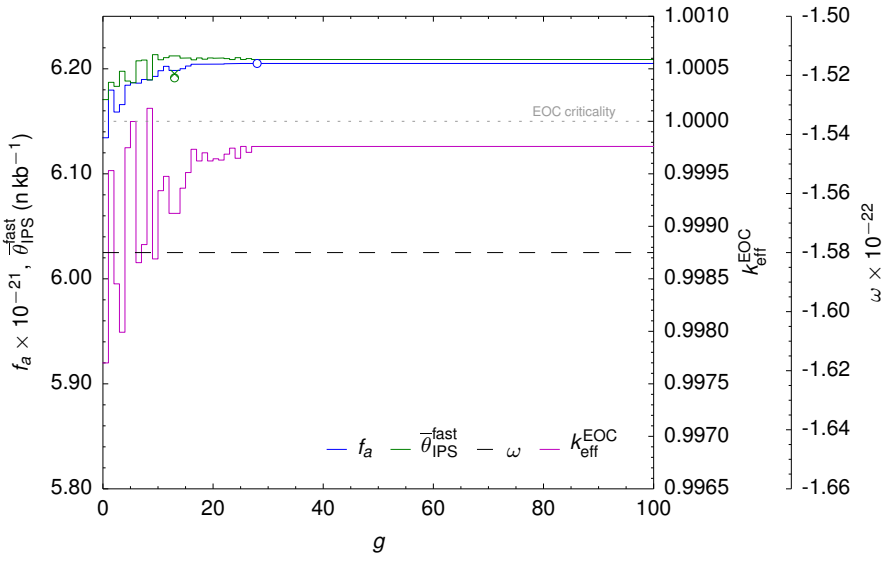


Figure 4.21: Problem 2 results (GA): evolution of the population-best fitness value (f_{a}) for $\omega = -1.58 \times 10^{22}$ (static penalty)

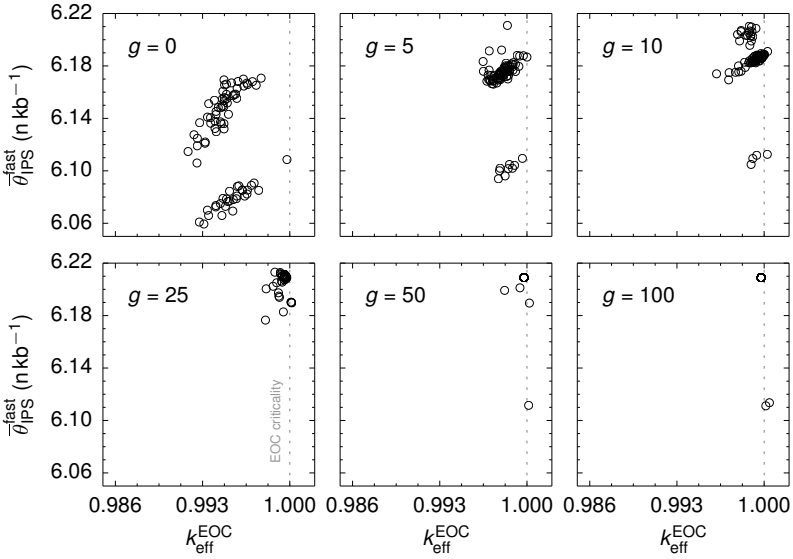


Figure 4.22: Problem 2 results (GA): evolution of the distribution of the objective function values ($\bar{\Theta}^{\text{fast}}_{\text{IPS}}$) for $\omega = -1.58 \times 10^{22}$ (static penalty)

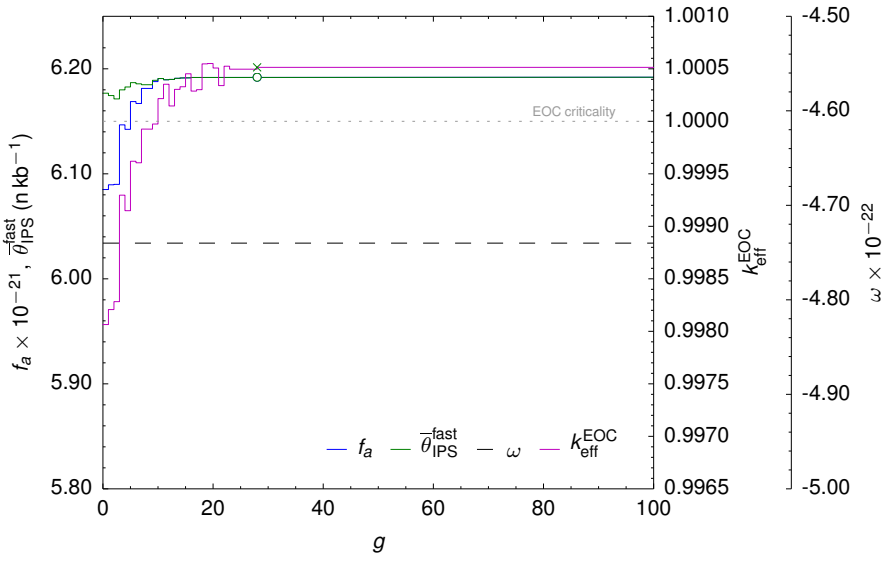


Figure 4.23: Problem 2 results (GA): evolution of the population-best fitness value (f_{a}) for $\omega = -4.74 \times 10^{22}$ (static penalty)

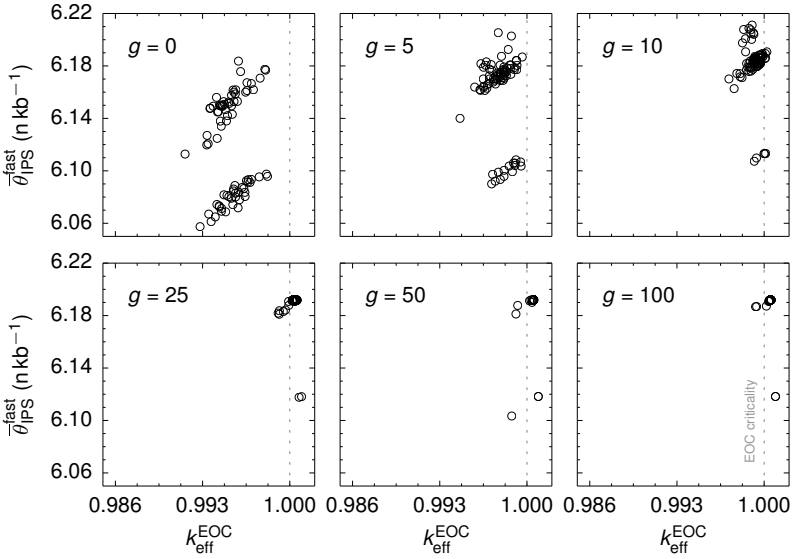


Figure 4.24: Problem 2 results (GA): evolution of the distribution of the objective function values ($\bar{\Theta}^{\text{fast}}_{\text{IPS}}$) for $\omega = -4.74 \times 10^{22}$ (static penalty)

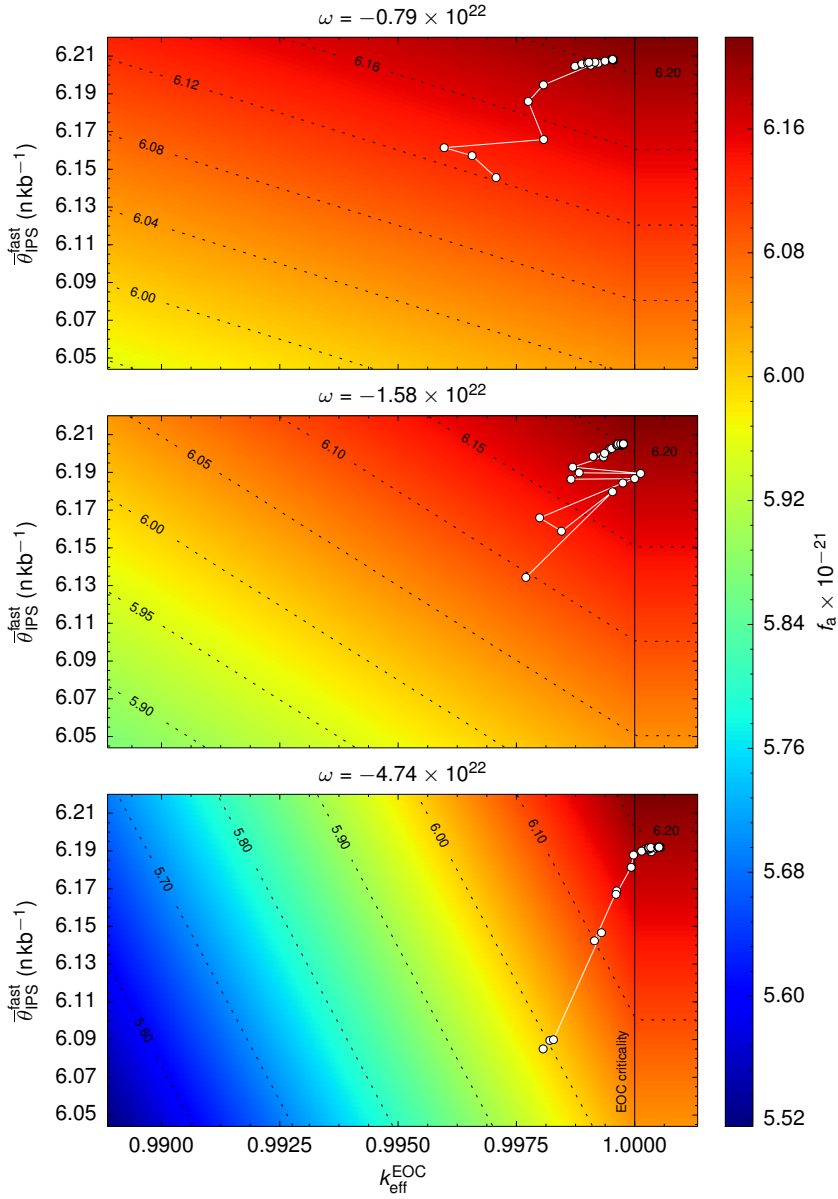


Figure 4.25: Problem 2 results (GA): evolution of the population-best individual in the f_{a}-landscape for three different ω values (static penalty)

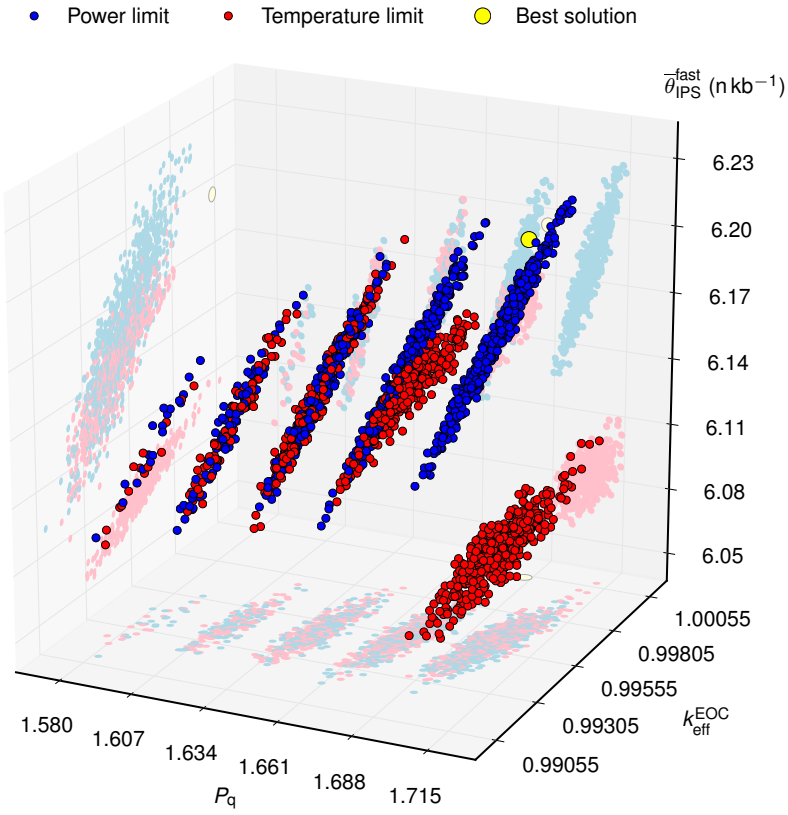


Figure 4.26: Distribution of random individuals in the P_{q}-$k^{\text{EOC}}_{\text{eff}}$-$\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ space

are gathered in two separate groups localized around the $\bar{\Theta}^{\text{fast}}_{\text{IPS}} = 6.15\,\text{n}\,\text{kb}^{-1}$ and $\bar{\Theta}^{\text{fast}}_{\text{IPS}} = 6.08\,\text{n}\,\text{kb}^{-1}$ values. The range of the $k^{\text{EOC}}_{\text{eff}}$ values is approximately the same for both groups. Further investigation of this phenomenon leads to a conclusion that the formation of both groups is a direct consequence of the discrete nature of the problem and of the method used for the evaluation of the individuals. This is better demonstrated in Figure 4.26, where 2000 randomly generated individuals are plotted in the P_{q}-$k^{\text{EOC}}_{\text{eff}}$-$\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ coordinates. One thousand of these individuals were generated using Approach 1, assuming the power limit $P = 100\,\text{MW}$ (blue points), and 1000 of them were generated using Approach 2, assuming the maximum cladding temperature limit $T_{\text{clad}} = 550\,°\text{C}$ (red points). The two clouds from Figure 4.22 and 4.24 correspond to the two clouds of the red points projected onto the $k^{\text{EOC}}_{\text{eff}}$-$\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ plane in Figure 4.26.

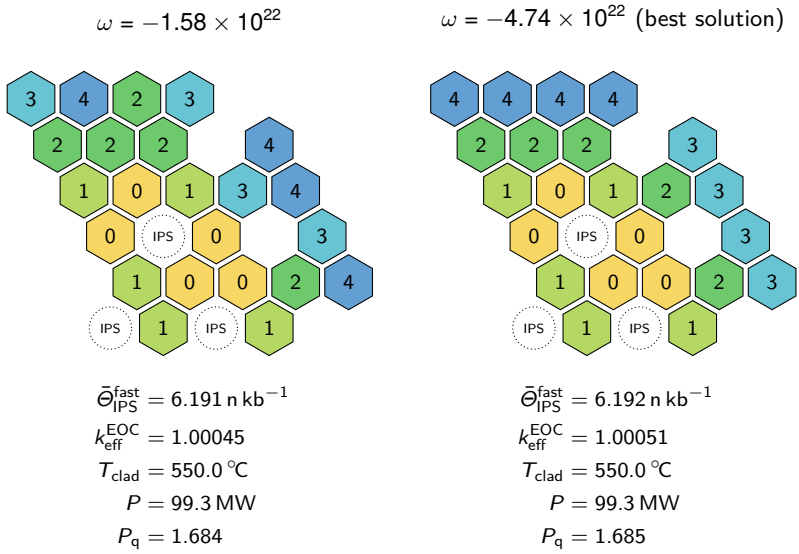


Figure 4.27: Problem 2 optimized solutions

The bottom cloud was formed by individuals with large power-peaking factors $P_{\text{q}} \gtrapprox 1.688$, for which the reactor power P, and thus also the neutron fluence $\bar{\Theta}^{\text{fast}}_{\text{IPS}}$, had to be lowered to satisfy the maximum cladding temperature constraint (4.12). The upper cloud is a partially distorted image of the original cloud represented by the blue points. It is also interesting to see that the randomly generated individuals are organized in five distinct swarms according to their P_{q} values, each of these swarms surrounding a different characteristic loading pattern. For instance, it is typical for the individuals from the right-most swarm that they contain at least three fuel assemblies of type 0 in the inner-most fuel-assembly ring; the individuals from this group with a large $\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ value have all six fuel assemblies in the inner-most ring of type-0, while the individuals with a low $\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ value have the other three fuel assemblies of type 4. The second swarm from the right has at least three fuel assembly of type 1 in the inner-most fuel-assembly ring, none of them being of type 0, and so on for the other swarms. In the left-most swarm, all fuel assemblies in the inner-most ring are of type 4. High $\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ values in this group are achieved when fuel assemblies of type 0 or 1 are in the second inner-most ring and low $\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ values are achieved if all 12 fuel assemblies present in the very center of the core are of type 4.

Both best individuals obtained as the results of the optimizations for $\omega = -1.58 \times 10^{22}$ and $\omega = -4.74 \times 10^{22}$ are depicted in Figure 4.27. It is believed

that the right loading pattern from the figure is also the global optimum of Problem 2.

Dynamic penalty. Several optimizations with different dynamic penalties, that is, with different ω_1 and ω_2 values from Eq. (4.25), were performed to solve Problem 2. In general, it can be concluded that the use of the dynamic penalty (4.25) did not lead to a better performing optimization algorithm. No better individual was found in a noticeably shorter time than when the static penalty was applied. Best solutions were received using the GA with a dynamic penalty factor $\omega(g)$, when this reached a large-enough value at an early stage of the optimization ($g \lessapprox 30$). This can be explained by the strongly convergent character of the GA, mainly attributed to the elitist generational replacement strategy. In other words, it is hard to obtain a better solution in the later stage of the optimization, since a vast majority of the population consists of the same or very similar individuals at that time.

Figures 4.28–4.31 present results obtained using two different dynamic penalties: one starting with a mild penalization determined by $\omega_1 = -1.58 \times 10^{22}$ and ending with a severe penalty determined by $\omega_2 = -15.8 \times 10^{22}$, and the other one starting with the severe penalty and ending with the mild one. It is apparent from Figure 4.28 and Figure 4.29 that while almost no population-best solutions were feasible in the first half of the optimization, more and more feasible solutions became to appear with the f_a receiving larger penalties in the second half of the optimization. An opposite tendency can be observed in Figure 4.30 and 4.31, where the f_a penalization was being steadily relaxed over the course of the optimization. Note that as soon as the value of the penalty factor dropped below a certain threshold limit ($\approx -2.5 \times 10^{22}$), the f_a could be further increased to the detriment of the $k_\mathrm{eff}^\mathrm{EOC}$ values, which fell below the criticality level immediately. A better solution was found in the latter case, which is equal to the right loading pattern from Figure 4.27.

Adaptive penalty. Similarly to the preceding cases with the dynamic penalty, an application of the adaptive penalty factor (4.26) also did not lead to any better solutions than in the case of the static penalty (because of the same reasons). Again, the penalization efficiency was eminently dependent on the initial value of the penalization factor $\omega(0, k)$. Also, lower values of the parameter k were preferred, so the $\omega(g, k)$ factor could swiftly react to the recent progress in optimization, before the algorithm converged. Figure 4.32 and Figure 4.33 show the results obtained for $\omega(0, k) = -3.16 \times 10^{22}$, where k was set to 5. It follows from the figure that the selection pressure toward larger $k_\mathrm{eff}^\mathrm{EOC}$ values was increased at the beginning of the optimization, only to be loosened shortly

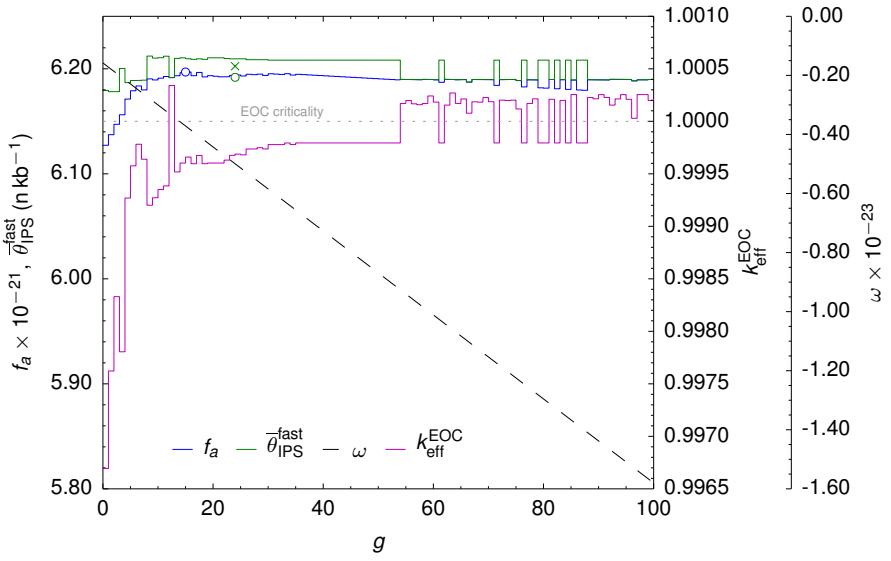


Figure 4.28: Problem 2 results (GA): evolution of the population-best fitness value (f_a) for $\omega_1 = -1.58 \times 10^{22}$ and $\omega_2 = -15.8 \times 10^{22}$ (dynamic penalty)

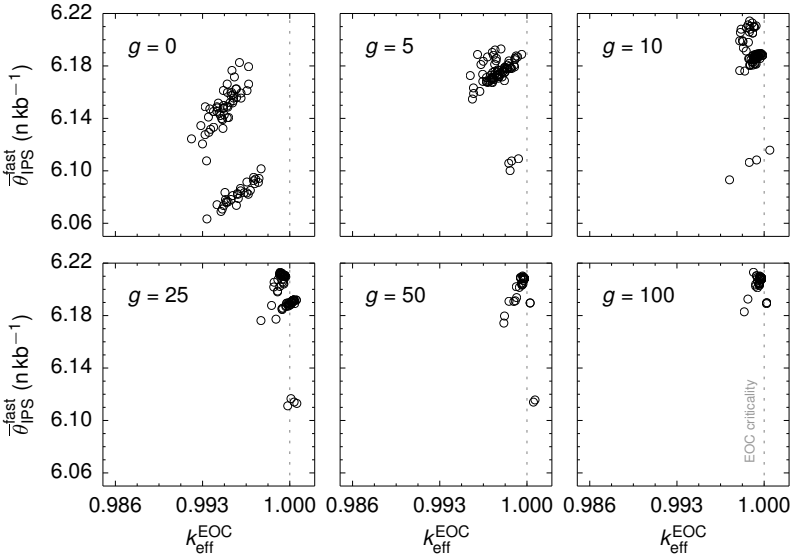


Figure 4.29: Problem 2 results (GA): evolution of the distribution of the objective function values ($\bar{\Theta}^{fast}_{IPS}$) for $\omega_1 = -1.58 \times 10^{22}$ and $\omega_2 = -15.8 \times 10^{22}$ (dynamic penalty)

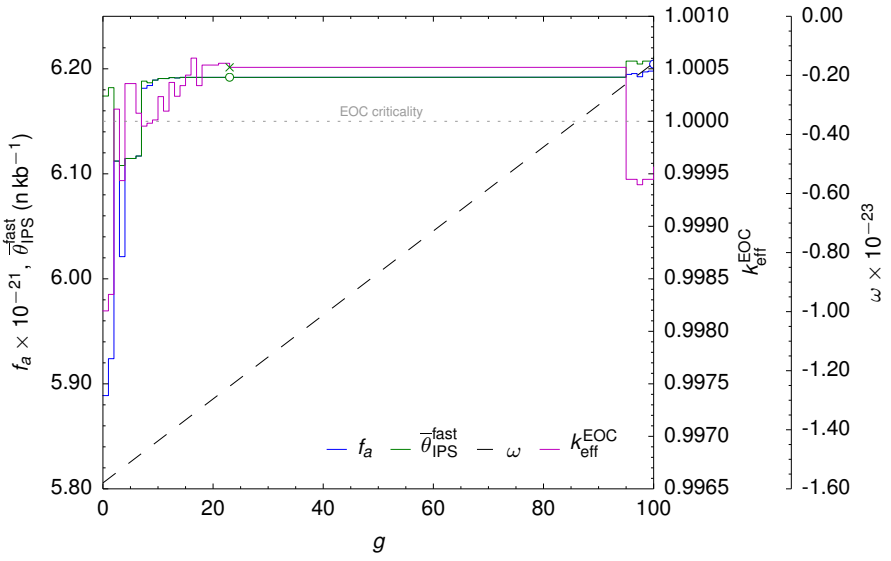


Figure 4.30: Problem 2 results (GA): evolution of the population-best fitness value (f_a) for $\omega_1 = -15.8 \times 10^{22}$ and $\omega_2 = -1.58 \times 10^{22}$ (dynamic penalty)

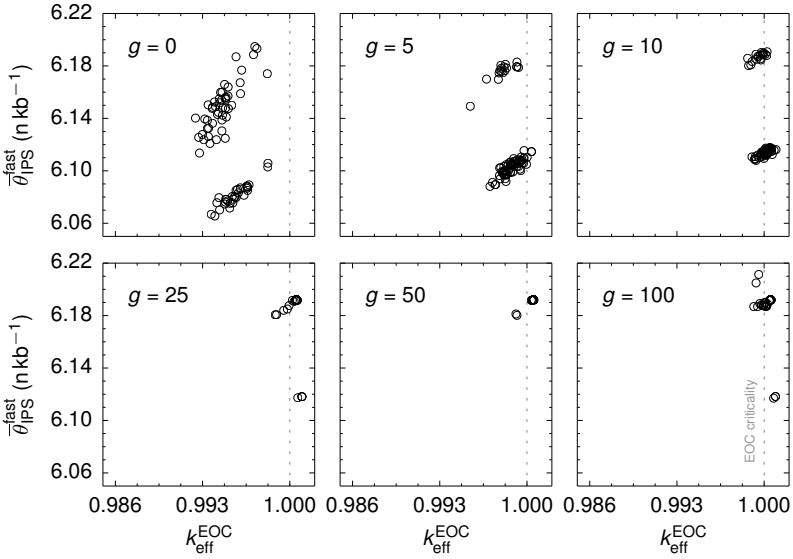


Figure 4.31: Problem 2 results (GA): evolution of the distribution of the objective function values ($\bar{\Theta}^{fast}_{IPS}$) for $\omega_1 = -15.8 \times 10^{22}$ and $\omega_2 = -1.58 \times 10^{22}$ (dynamic penalty)

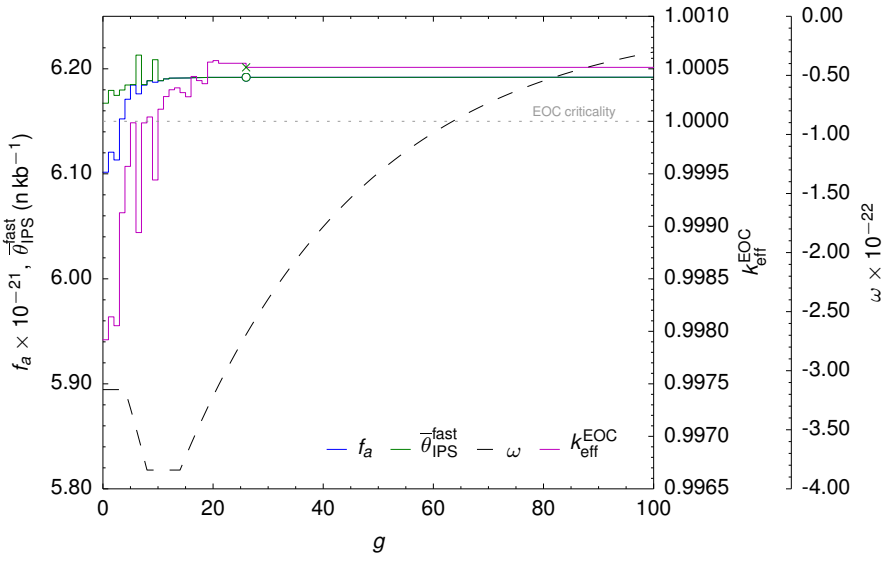


Figure 4.32: Problem 2 results (GA): evolution of the population-best fitness value (f_{a}) for $\omega(0,k) = -3.16 \times 10^{22}$ and $k = 5$ (adaptive penalty)

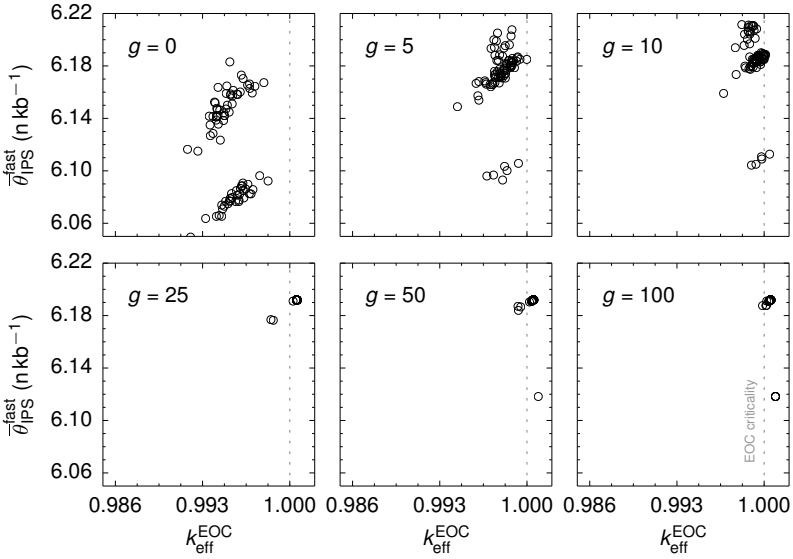


Figure 4.33: Problem 2 results (GA): evolution of the distribution of the objective function values ($\bar{\Theta}^{\text{fast}}_{\text{IPS}}$) for $\omega(0,k) = -3.16 \times 10^{22}$ and $k = 5$ (adaptive penalty)

afterward, when the algorithm began to give feasible solutions consistently. The best solution from Figure 4.27 was found again with the adaptive penalty.

The results of solving Problem 2 using the GA with different variants of penalization are summarized in Table 4.5.

ACO Results

The ACO algorithm with a population of 30 ants was used to solve Problem 2. The number of iterations was increased to $I = 268$ in this case. As such, the maximum possible number of solution evaluations was approximately the same as in the case of the GA[50]. The value of the parameter γ in the global updating rule (4.7) was set to 3.6×10^{-22} for all calculations.

The results obtained by applying the ACO algorithm to Problem 2 are summarized in Table 4.6 for all three types of penalizations. The penalty factors used in the calculations are also indicated in the table. It was observed that better results were obtained by applying the dynamic and adaptive penalties rather than the static penalties[51], which performed best for the GA (cf. the results from Table 4.5). This observation is in correspondence with the general expectation that the penalization technique with a varying penalty factor would perform better for a slowly-convergent optimization method like ACO. The optimum solution was found after 117 iterations, when the dynamic steadily increasing penalization was used (#3 in Table 4.6), and after 211 iterations, when the adaptive penalization was used (#5). A comparable number of candidate solutions were evaluated in both cases.

The graphical representation of the results obtained for the adaptive penalty is given in Figure 4.34 and Figure 4.35. It can be seen from the first figure that the penalization was progressively decreasing in a step-wise fashion during the optimization. While most of the solutions were belonging to the cloud of solutions characterized by lower $\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ values around iteration 10, these moved quickly into the cloud of larger $\bar{\Theta}^{\text{fast}}_{\text{IPS}}$ values (see the plot for $i = 25$ in Figure 4.35), due to the stronger penalization at the beginning of the optimization. This also led to a successive series of feasible iteration-best solutions, allowing for a decrease in the penalization between the iterations ≈ 50 and ≈ 70. Once a new successive series of $k = 25$ feasible solutions was reestablished for the new value of the $\omega(i, k)$ factor, this began to be loosened again around the 120th iteration

[50] This number was $80 \times (100 + 1) = 8080$ for the GA and slightly less for the ACO: $30 \times (268 + 1) = 8070$.

[51] A more comprehensive study employing statistical techniques would be needed to validate this conclusion.

Table 4.5: Problem 2 results (GA)

#	Penalization			Maximum f_a (fitness)					Maximum feasible $\bar{\Theta}_{IPS}^{fast}$				NE[c]
	Type[a]	$\omega(0,k)$, ω_1, $\omega \times 10^{-22}$	ω_2 $\times 10^{-22}$	f_a $\times 10^{-21}$	$\bar{\Theta}_{IPS}^{fast}$ (n kb^{-1})	k_{eff}^{EOC}	g	FS[b]	$\bar{\Theta}_{IPS}^{fast}$ (n kb^{-1})	k_{eff}^{EOC}	g	FS[b]	
1	S	−0.79	–	6.2082	6.2119	0.99953	22	0	6.1904	1.00005	54	25	1834
2	S	−1.58	–	6.2051	6.2089	0.99976	28	1	6.1914	1.00045	13	42	2220
3*	S	−4.74	–	6.1920	6.1920	1.00051	28	91	6.1920	1.00051	28	92	2230
4	D	−1.58	−15.8	6.1970	6.2096	0.99966	15	39	6.1919	1.00052	24	86	3396
5*	D	−15.8	−1.58	6.2045	6.2114	0.99956	100	90	6.1920	1.00051	23	98	2060
6*	A[d]	−3.16	–	6.1920	6.1920	1.00051	26	92	6.1920	1.00051	26	93	2030

[a] Static (S), dynamic (D), and adaptive (A).
[b] Number of population-best feasible individuals encountered during $G = 100$ iterations.
[c] Number of evaluations.
[d] $\beta_1 = 1.05$, $\beta_2 = 1.03$, and $k = 5$ in Eq. (4.26) for the adaptive penalty.
* Best result ever found obtained.
$\alpha = 1$ in Eq. (4.23) for all types of penalizations.

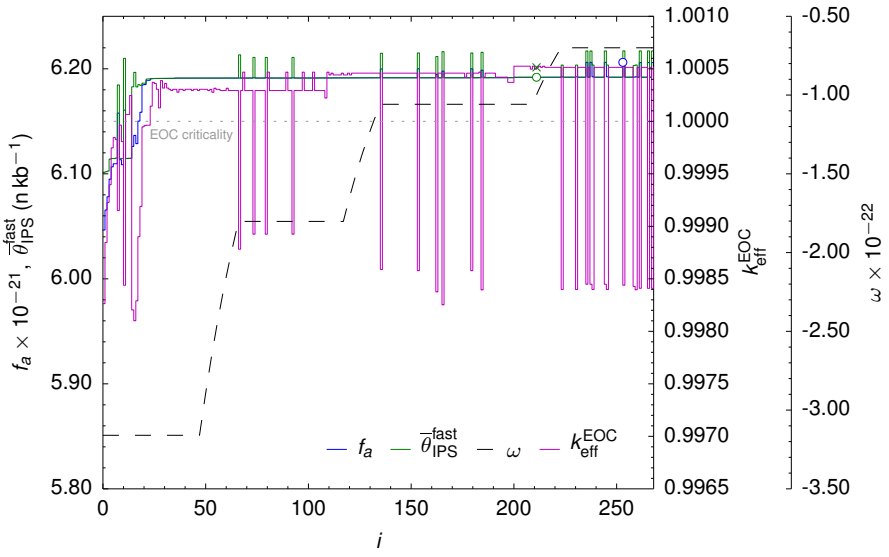


Figure 4.34: Problem 2 results (ACO): evolution of the iteration-best f_{a} value for $\omega(0,k) = -3.16 \times 10^{22}$ and $k = 25$ (adaptive penalty)

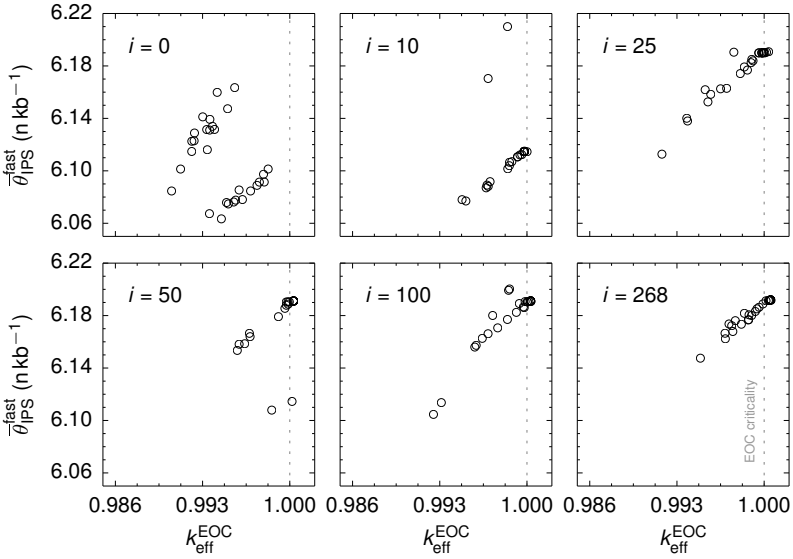


Figure 4.35: Problem 2 results (ACO): evolution of the distribution of the objective function values ($\bar{\Theta}^{\text{fast}}_{\text{IPS}}$) for $\omega(0,k) = -3.16 \times 10^{22}$ and $k = 25$ (adaptive penalty)

Table 4.6: Problem 2 results (ACO)

#	Penalization			Maximum f_a					Maximum feasible $\bar{\Theta}_{IPS}^{fast}$				NE[c]
	Type[a]	$\omega(0,k)$, ω_1, $\omega \times 10^{-22}$	$\omega_2 \times 10^{-22}$	f_a $\times 10^{-21}$	$\bar{\Theta}_{IPS}^{fast}$ $(n\,kb^{-1})$	k_{eff}^{EOC}	i	FS[b]	$\bar{\Theta}_{IPS}^{fast}$ $(n\,kb^{-1})$	k_{eff}^{EOC}	i	FS[b]	
1	S	−4.74	–	6.1881	6.1881	1.00018	195	89	6.1881	1.00018	195	119	4410
2	S	−7.90	–	6.1919	6.1919	1.00033	132	187	6.1919	1.00033	132	195	4316
3*	D	−1.58	−15.8	6.1920	6.1920	1.00051	117	202	6.1920	1.00051	117	208	4434
4	D	−15.8	−1.58	6.1915	6.1915	1.00046	245	259	6.1915	1.00046	245	263	3966
5*	A[d]	−3.16	–	6.2062	6.2171	0.99844	253	226	6.1920	1.00051	211	255	4174

[a] Static (S), dynamic (D), and adaptive (A).
[b] Number of iteration-best feasible solutions encountered during $I = 268$ iterations.
[c] Number of evaluations.
[d] $\beta_1 = 1.05$, $\beta_2 = 1.03$, and $k = 25$ in Eq. (4.26) for the adaptive penalty.
* Best result ever found obtained.
$\alpha = 1$ in Eq. (4.23) for all types of penalizations.

and the same procedure was repeated two times more before the optimization ended.

It is concluded that the ACO algorithm used here for solving Problem 2 was able to find the best solution ever found for this problem, depicted in Figure 4.27. Comparing to the elitist GA, approximately twice as many candidate solution evaluations were needed for that, which favors the GA over the ACO.

4.5 Summary

Two different discrete loading pattern optimization problems were solved in this chapter for the critical MYRRHA-FASTEF core, using two different metaheuristic optimization methods and the reactor core models developed in Chapter 3. The first problem was defined as a simple test problem with a known optimum solution and it was used to study the properties of the adopted optimization methods. The goal was to find a maximum value of the BOC effective multiplication factor k_{eff}, given a set of available fuel assemblies of five different types. The second problem was proposed as a more realistic MYRRHA loading pattern optimization problem, aimed at maximization of the average IPS fast-neutron fluence. Two additional constraints were considered in this case, besides the fuel inventory constraint from Problem 1: the maximum fuel-cladding temperature constraint and the EOC criticality constraint. Both problems were formulated assuming all-rods-out conditions and the one-third core symmetry known from Chapter 3.

The two metaheuristics selected for solving the MYRRHA core loading problems were the Genetic Algorithm (GA) and Ant Colony Optimization (ACO). Both methods belong to nature-inspired evolutionary search techniques and have been applied to similar problems before. The chapter contains a detailed description of both algorithms, including an elaborate discussion of the algorithm components. A parametric study was done for each of the algorithms applied to Problem 1, in order to identify parameter values that improve their performance. In this way it was found, for instance, that elitism can significantly increase the efficiency of the search, if embedded in the GA. The results of the studies were used later for solving Problem 2.

Different techniques were used to deal with the three constraints considered in Problem 2. The limited number of available fuel assemblies of different types was taken into account when generating/constructing the solutions to the problem, before they were evaluated. A special repair operator was introduced into the GA for this reason. Another repair mechanism was used in both algorithms to

treat the maximum cladding-temperature constraint. In order to make every trial loading pattern meet this constraint, a special procedure was implemented in the algorithms that, if needed, modulates the reactor power and lowers the maximum cladding temperature accordingly. Finally, three variations of the classical objective-function penalization technique were used to handle the EOC criticality constraint: static, dynamic, and adaptive penalizations.

The obtained optimization results proved that both algorithms can deliver high-quality solutions to the problems, which comply with all imposed constraints. The optimum solution was found successfully for Problem 1 and it is believed also for Problem 2. These solutions are depicted in Figure 4.8 and Figure 4.27, respectively. The largest success rate in finding the optima and the lowest number of required candidate solution evaluations were achieved for the GA with the elitist population replacement strategy. This makes it the favored optimization method for solving the MYRRHA fuel loading problems. Around 2000 loading-pattern evaluations were needed to find the Problem 2 best solution, when using the GA, and approximately twice as many evaluations, when using the ACO. This corresponds to less than 2.5 hours, respectively, 5 hours of computing time, assuming the use of the reference computing platform described in this work, easily fulfilling the expectations regarding the maximum time limits for one MYRRHA loading-pattern optimization. A simple parallelization of both population-based algorithms can further significantly reduce these computing times.

Chapter 5

The RELOAD-M Tool

The in-core fuel management objectives discussed in Chapter 2 and more specifically in Section 2.1.2 are usually attained by implementing some sort of a systematic procedure in the in-core fuel management decision-making process. In the case of the loading pattern optimization problem, these often have a form of a software application, which is able to find fuel-loading configurations that perform well in terms of fulfilling the objectives, while meeting all safety, operational, and other constraints at the same time. In the nuclear power industry, a certain type of such an application is present at every company providing in-core fuel management services to electric utilities or at the utilities themselves. The level of its sophistication may vary from a simple scoping analysis tool, which should help a nuclear engineer to ease the process of designing new core loadings, typically by trial-and-error, or a simple search method, to an advanced system that can generate highly optimized solutions to the problem with a minimal engineer's input. Similar in-core fuel management tools are less (if at all) used at research facilities or at other types of nuclear facilities that have a different primary purpose than to produce electric power. A sample list of existing software applications, which were mentioned in scientific papers and used for solving the loading pattern optimization problem for various types of nuclear facilities, is included in Table A.1 in the appendix.

Despite the fact that many in-core fuel management tools have been developed for tackling the core reload problem, none of them could be used directly for MYRRHA for several reasons. For instance, every such tool is generally tailored to a nuclear reactor of a specific type, design, purpose, etc. This means that only a limited number of objectives and constraints of a certain type can be handled by the tool. Also, the use of a specific nuclear reactor analysis

code tightly coupled to the in-core fuel management tool further restricts the tool's applicability to a particular type of nuclear reactor (e.g., a PWR with a rectangular fuel-pin lattice). Other important aspect is that the in-core fuel management tools are mostly proprietary and only a very few commercial software packages are available in this area, which are all dedicated to light-water power reactors[1]. For these reasons and for the reasons given in Section 1.1.4, a special tool named RELOAD-M (REactor Loading Optimization And Design for MYRRHA) was developed during this PhD project that can solve the loading pattern optimization problem for MYRRHA, taking into account the special purpose and character of the facility. Chapter 5 contains a general description of RELOAD-M, including its functionality, high-level architecture, component design, and usage. Possible future developments in RELOAD-M are suggested at the end of the chapter.

5.1 Overview

5.1.1 Functionality

RELOAD-M is a versatile in-core fuel management software tool that was developed primarily to solve the single-cycle MYRRHA loading pattern optimization problem. The design of the tool makes it also possible to perform a scoping analysis for any user-specified MYRRHA loading pattern, as long as it can be described by the incorporated reactor-physics models. The MYRRHA-FASTEF critical reactor core design described in Section 2.4 was considered in this dissertation. As an in-core fuel management tool, RELOAD-M evaluates the loading patterns in terms of their physics characteristics (neutronics and thermal-hydraulics). No economic assessment is performed.

5.1.2 Features

RELOAD-M's versatility lies in its two distinct features: the tool's capability to cope with different reactor analysis codes and the tool's capability to cope with different optimization methods for tackling the core-reload problem.

[1]To name some examples: the ROSA code system for the PWR LP optimization (NRG, 2014; Verhagen and Wakker, 2003), the XIMAGE optimization suite for both PWRs and BWRs (Stevens and Rempe, 2003; Studsvik, 2014), or the ALPS code for PWRs (Bradfute et al., 1997; Johansen, 1994; Shatilla et al., 2000).

Reactor Analysis Codes

The first feature gives the user the possibility to consider various loading-pattern characteristics, when formulating the optimization objectives and constraints. Three different reactor analysis codes are currently available in RELOAD-M: the DIF3D10.0 steady-state neutronics code; REBUS-PC 1.4, the DIF3D extension with the depletion option; and a simple thermal-hydraulics module for calculating the maximum cladding temperature. All reactor codes with the corresponding MYRRHA-FASTEF reactor-physics models are described in detail in Chapter 3 and Appendix B. They allow to estimate the following quantities for every MYRRHA-FASTEF core configuration at both BOC and EOC states: the effective multiplication factor k_{eff}, the per-node per-energy-group averaged neutron flux and power-density distributions $\bar{\phi}_g^k$ and $\bar{q}_g^k$, the maximum achieved linear power $q_{\max}$, and the maximum achieved cladding temperature T_{clad}.

The tool's modeling capacity can be further extended by modifying the present reactor-physics models or by coupling other reactor analysis codes to it. This way, RELOAD-M can be applied to other MYRRHA core designs characterized by a generally different core layout (i.e., number and positions of FAs of different types, CR and SR arrangement), power level, core-components design, fuel properties, etc. In this sense, the number and type of different reactor-core characteristics that can be spanned by RELOAD-M is limited only by the modeling capabilities of the adopted reactor analysis codes. Ultimately, RELOAD-M can be applied to any reactor type if adequate modeling tools are coupled to it.

Optimization Methods

The possibility to choose for different optimization methods, on the other hand, allows the user to opt for a solution method that performs best for a particular problem with respect to the quality of the results or the needed optimization time. Additionally, a more academic motivation for adding this feature to the tool was to make it possible to compare different optimization methods when applied to the same core-reload problem.

Essentially any optimization method relying on the use of reactor codes (see Section 2.2.4) can be included in RELOAD-M. The two metaheuristic optimization methods currently implemented in the tool, the rank-based GA and ACO, have been described in Chapter 4. Both methods exploit a vector representation of the solutions (LPs) and can solve the MYRRHA-FASTEF loading pattern optimization problem for different reactor-core symmetries

depicted in Figure 4.5. The GA also allows to utilize "best-guess" solutions during the algorithm initialization, which helps to speed up the optimization process. Best-guessed solutions are user input based on engineering judgment and experience feedback.

The optimization objectives and constraints can be any function of the loading-pattern characteristics listed in the preceding paragraph. The following functions can be calculated by RELOAD-M and used for the optimization besides k_{eff}, $q_{\max}$, and T_{clad}; again, for both BOC and EOC conditions: the total, radial, and axial power-peaking factors P_{q}, P_{r}, and P_{z}, and the maximum and average total, thermal, and fast neutron flux in a selected core region (e.g., the central IPS channel or the region consisting of all IPS nodes in the fissile zone). Some additional functions were used in Chapter 4, also included in the current version of RELOAD-M: the EOC multiplication factor $k_{\mathrm{eff}}^{\mathrm{EOC}}$ estimated according to Eq. (4.22) and the average total, thermal, and fast neutron fluence in the IPS region $\bar{\Theta}_{\mathrm{IPS}}$ calculated either for a fixed reactor power P or for the decreased power P from Eq. (4.22).

5.1.3 Implementation

The RELOAD-M code is organized in three distinct parts (components): the main part, which provides a communication interface to the user and also a standardized (communication) data-exchange interface to the two other parts of the tool—the optimization algorithm and the reactor code. The global structure of the tool is shown in Figure 5.1 and a closer description of all three components is given below in Section 5.2.

RELOAD-M was written in Python under the 64-bit Linux operating system Ubuntu (Canonical Ltd., 2014). Python (Python Software Foundation, 2014) is a popular cross-platform scripting object-oriented programming language with a highly-readable syntax, advanced memory management, and a large number of available libraries, making it also the programming language of choice for this project. The fact that it is a scripting programming language was considered unimportant, since the implemented optimization algorithms are simple and as such impose only a minimum computational burden comparing to the employed computationally much more expensive reactor analysis codes. The Python 3 version of the language was used in this work.

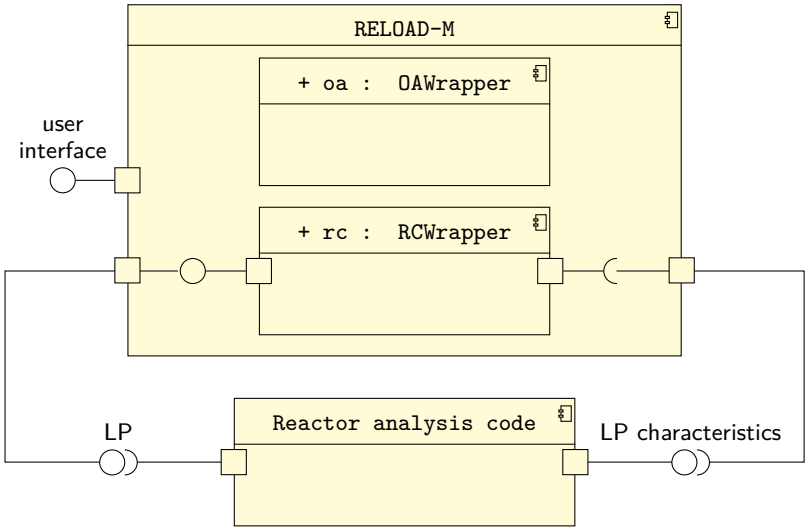


Figure 5.1: RELOAD-M component diagram

5.2 Design

This section gives a description of the principal RELOAD-M components from the component diagram 5.1. The basic idea behind this scheme was to develop a software application for solving the core-reload problem that can use different optimization algorithms and reactor codes and can be easily extended with new instances of both kinds. This was achieved by encapsulating both algorithms and reactor codes into special wrappers, providing them with a unified interface. More detailed information on the system design follows from the class diagram 5.2, which presents the main classes of all three components and the relations between them[2]. As the diagram suggests, multiple instances of the `OAWrapper` and `RCWrapper` classes can be associated with the main class named `Tool`, but only one instance of each class is actually used for the optimization, which is selected by the user (see the `oa` and `rc` attributes of `Tool`).

[2]Both diagrams, the component diagram 5.1 and the class diagram 5.2, were prepared using the Unified Modeling Language (UML 2) notation and semantics (Object Management Group, 2014).

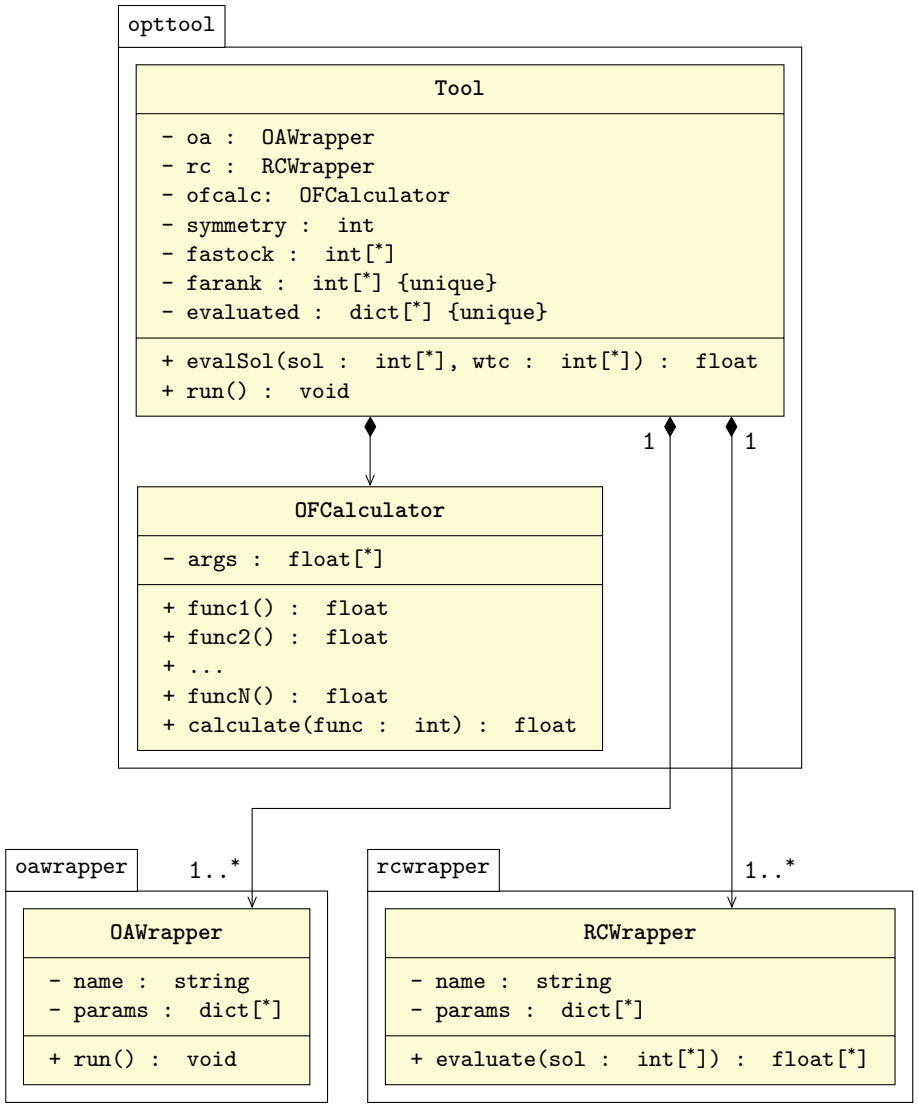


Figure 5.2: RELOAD-M simplified high-level class diagram

5.2.1 Main Component

The main component RELOAD-M from Figure 5.1 corresponds to the `opttool` module (package) from Figure 5.2. This module contains the source code and the definition of all classes that are responsible for the most important functions of the tool: reading input files, providing a graphical user interface (GUI) to the user, controlling the optimization process, evaluating objective functions, and generating an output. Diagram 5.2 shows the two most important classes of the module[3]: the main class `Tool`, which is instantiated when RELOAD-M is started, and the class `OFCalculator`, which is used to calculate the objective-function value for the selected function `func` and for given values of the function arguments `args`. For instance, the $\bar{\Theta}_{\mathrm{IPS}}^{\mathrm{fast}}$ values were calculated by `OFCalculator` in Problem 2 from Section 4.4.2 for every loading pattern evaluated by the reactor code.

The `Tool` class holds all information needed for a successful execution of the optimization algorithm, including the type of applied optimization algorithm (`oa`), the type of applied reactor code (`rc`), the type of objective function (`func`), the type of solution symmetry (`symmetry`), the number of different types of available fuel assemblies (`fastock`) and their ranks (`farank`), and the list of all unique loading patterns evaluated so far (`evaluated`). All relevant information is available to the optimization algorithm and reactor code throughout the whole optimization process.

`Tool` also provides a communication layer between the optimization algorithm and the reactor code. Its role as a mediator between the two objects can be explained by describing the logical flow in which their mutual interactions occur as the optimization algorithm proceeds through iterations. Every time a new solution `sol` is generated by the optimization algorithm, the `oa` object passes this solution to the `rc` object as an argument of the `Tool`'s `evalSol()` function. Then, the solution is evaluated by the reactor code and the obtained solution characteristics are processed further by the `evalSol()`, which uses them for the calculation of the final objective-function value associated with the solution. This value is then passed back to `oa`, and the whole sequence is repeated for the next solution and so on, until the optimization algorithm finishes execution. In the standard notation used in this text, `sol` is equal to the loading pattern $\boldsymbol{x}$, the selected solution characteristics `wtc` returned by `rc` are denoted as $\boldsymbol{y}$, and the `evalSol()` function corresponds to the objective function f or to the augmented function f_{a}. In the GA and ACO from Chapter 4, `evalSol()` substitutes the `Evaluate()` function from Algorithm 4.3. Here, the

[3]It should be noted though that the module `opttool` contains some other classes as well, which are mostly used for the definition of different GUI components, graphical representation of the results, etc.

function f from line 7 of the algorithm corresponds to one of the `OFCalculator` functions `func1()`, `func2()`, ..., `funcN()`, and the memory $\mathcal{M}$ corresponds to the `evaluted` attribute.

In order to make this scheme work, it is important that both subcomponents use the same data representation of a solution. In other words, it is important to know, before the wrappers are implemented, how to translate a solution from the form recognized by the optimization algorithm to the form recognized by the reactor code. In the particular case of the `OAWrapper` and `RCWrapper` classes implemented in this work, the simple vector representation with the vector-to-LP mapping described in Section 4.3.1 and depicted in Figure 4.2 were used.

5.2.2 Optimization-Algorithm Component

The second core component of the RELOAD-M tool is the `OAWrapper` component represented by the module and the class of the same name in the class diagram 5.2[4]. An instance `oa` of this class acts in the tool as the optimization algorithm applied to the problem. It is denoted by the attribute `name` and launched by calling the `run()` function. The attribute `params` stands for the list of all parameters that can be supplied to the algorithm (e.g., the input parameters listed in the top of the ACO algorithm 4.10). These parameters are passed to `oa` by an instance of the `Tool` class, which gets them from the input file or from the interactive user interface.

The use of the word "wrapper" in the name of the class indicates that `OAWrapper` may also act as an interface to some type of a preexisting third-party software application, usable for solving the loading pattern optimization problem, rather than as the optimization algorithm itself. This allows to utilize different general-purpose optimization frameworks in RELOAD-M, making use of most of the benefits such a solution offers, like, for instance, reduced implementation effort achieved by a maximum reuse of an error-free highly-optimized code, or the ability to apply various techniques and variants with little additional effort[5].

Both approaches were used in this work when implementing the `OAWrapper` class: the ACO algorithm described in Section 4.3.2 was implemented completely in Python, whereas the GA from Section 4.3.1 was implemented exploiting

[4]Besides the main `OAWrapper` class, the `oawrapper` module typically contains other classes to implement the optimization algorithm.

[5]General aspects concerning the optimization frameworks for metaheuristics were described by Talbi (2009) in his book. More comprehensive and updated information can be found in the survey by Parejo et al. (2011), who reviewed and compared practically all metaheuristics optimization frameworks that were available at the time the survey was written.

Paradiseo, a "white-box" template-based C++ library dedicated to a flexible design of EAs (Paradiseo team, 2014; Talbi, 2009)[6]. Both algorithms are described in the algorithm listings 4.1–4.12. Though somewhat simplified, these listings preserve well the logical flow of both algorithms. The fuel-inventory vector $\boldsymbol{m}$ used in the algorithms is represented by the `fastock` attribute in the `Tool` class and the fuel-type vector $\boldsymbol{t}$ by the `farank` attribute. As was already said, all other parameter values used in the descriptions, such as the population size p in case of the GA or the initial pheromone amount τ_0 in case of the ACO, are processed by `Tool` and stored in the `OAWrapper.params` attribute. Other parameter values involved in the calculations of the objective function, such as the initial value of the penalization factor ω, are managed exclusively by `Tool` and do not enter `OAWrapper`.

5.2.3 Reactor-Code Component

As follows from Figure 5.1, the `RCWrapper` component serves as an interface that provides a communication channel between the main component `RELOAD-M` and the reactor analysis code component. It is implemented as the `RCWrapper` class from Figure 5.2. Similarly to `OAWrapper`, each `RCWrapper` instance is identified by its `name`. The values of all parameters needed for the execution of the particular code are stored in the attribute `params`. An example of such a parameter can be the type of the symmetry to be employed in the evaluation of a loading pattern `sol`. Calling the `evaluate()` function runs the reactor code for the supplied core configuration `sol`.

The `evaluate()` function implemented in this work performs the following actions: first, it generates an input file for the given loading pattern `sol` and parameter values `params` provided by the `Tool` object; then, it executes the reactor code (e.g., DIF3D or REBUS); and finally, it processes the output file and returns the extracted loading-pattern characteristics back to the `Tool` object. The `evaluate()` function is called from the `Tool.evalSol()` function. The whole sequence is depicted in Algorithm 4.3 in lines 3–5.

5.3 Usage

This section gives a short overview of some practical aspects of using the RELOAD-M tool by a user.

[6]The interoperability between the Paradiseo C++ code and the `OAWrapper` Python code was provided by the portable Boost.Python C++ source library (Abrahams, 2014).

5.3.1 Input

The RELOAD-M input file has a form of an Extensible Markup Language (XML) (W3C, 2014) document and contains the following information:

- type of the optimization algorithm to be used for the optimization,
- type of the reactor code to be used for the optimization,
- specification of the objective function to be maximized/minimized,
- type of the optimization (maximization/minimization),
- type of the reactor-core symmetry assumed in the problem to be solved,
- description of the part of the reactor core that is subject to change, including the identification of the loading-pattern positions represented by the vector $\boldsymbol{x}$ and closer specification of all other "static" positions (CR positions, SR positions, etc.),
- ranks of all different fuel-assembly types,
- number of available fuel assemblies of the different types,
- various parameters to be supplied to the selected optimization algorithm (e.g., the size of the population in the GA),
- and various parameters to be supplied to the selected reactor code.

5.3.2 Output

The current version of RELOAD-M generates two types of output: the standard output stream, which is forwarded during the RELOAD-M execution to a special GUI window and can be saved at the end of the optimization as a text file; and a set of XML files, two files per evaluated loading pattern. One XML file contains a complete description of the layout of the loading pattern and the other file contains a record of all its characteristics, as they were calculated by the reactor code. The first type of output includes the same data supplemented by some additional information that can be used for debugging purposes.

5.3.3 User Interface

A simple graphical user interface (GUI) was developed for RELOAD-M, based on PyQt (Riverbank Computing Limited, 2014), a set of Python bindings for

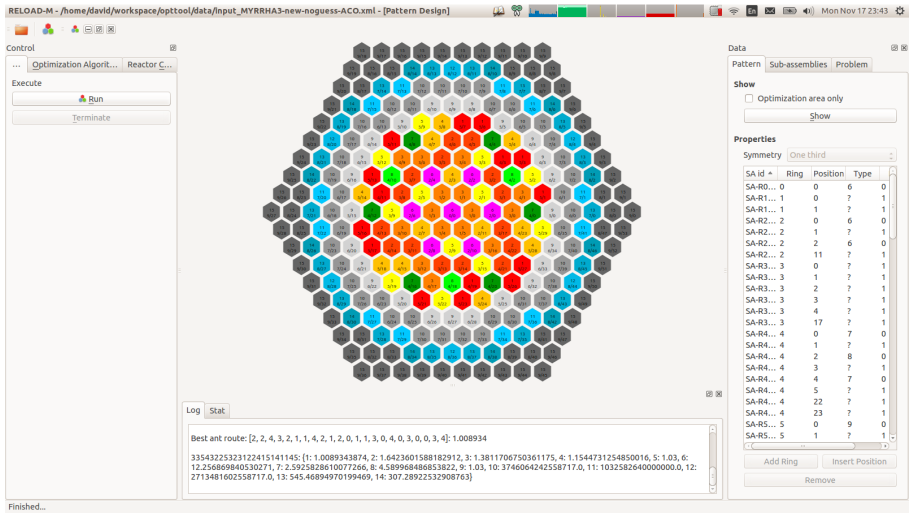


Figure 5.3: RELOAD-M graphical user interface

the Qt application user-interface framework (*Qt Project* 2014). A screenshot of the main RELOAD-M GUI window is shown in Figure 5.3. It serves the purpose of the only user interface provided by RELOAD-M and allows the user to:

- select and open an input file,
- modify the values of all parameters parsed from the input file,
- initiate, monitor, and terminate the optimization process,
- and display the best obtained loading pattern, once the optimization is finished.

5.4 Future Developments

Although the current version of RELOAD-M can already deliver solutions to some MYRRHA-FASTEF loading pattern optimization problems, the development of the tool is still at its beginning and a lot of work remains to be done, before it will reach a high level of maturity. Significant effort should be dedicated especially to continuous improvement of the source code, its organization and optimization. For instance, one of the obvious improvements, which can effectively lower the computing times typical for the currently implemented population-based

optimization methods, is the parallelization of the code (e.g., Bolloni et al., 1999; Cantú-Paz, 1998; Hays, 2009; H. C. Lee et al., 2001; Norouzi et al., 2013; Schirru, Lima, et al., 2006; Waintraub et al., 2009; Yamamoto and Hashimoto, 2002). The implementation of an advanced mechanism (database) for storing data for all solutions ever evaluated would be also useful and could lead to better overall optimization performance of the tool.

Additional attention could also be given to increasing the tool's functionality. One of the features that would improve the tool's usability much, would be, for instance, a visual and interactive way of designing MYRRHA loading patterns to be evaluated afterward by a reactor analysis code. Another function that would substantially extend the field of application of RELOAD-M is the capability to solve multicycle loading pattern optimization problems.

Two separate categories of possible improvements that deserve special attention, but are not discussed here, are improvements in the optimization methods and improvements in the reactor analysis codes. Some future perspectives in these two areas are drafted in the Conclusion chapter.

5.5 Summary

The RELOAD-M in-core fuel management tool was developed during the PhD project with the goal to solve the MYRRHA loading pattern optimization problem. It exhibits some characteristics that make it well suited for a multipurpose research facility with a broad range of applications. The tool allows to use different reactor codes for the evaluation of the MYRRHA loading patterns, which means that various objectives and constraints can be considered for the MYRRHA core-reload problem. Another distinct characteristic is that multiple optimization methods can be employed by the tool to solve the problem at hand.

All of this was made possible by adopting a high-level design of the tool, which consists of three components interconnected via well-defined interfaces. The main component that provides a user interface to the user and passes data between the optimization algorithm and the reactor code, both being encapsulated in the two other components. The design and implementation of all three parts has been described in this chapter with references to the GA and ACO algorithms presented in Section 4.3.1 and 4.3.2.

RELOAD-M works with XML input files that consist of a complete description of the optimization problem and parameter values used by the optimization algorithm and reactor code of choice. Output is also generated in the form of

XML files (besides the standard output printed to the screen) and contains a description of all evaluated loading patterns, including their characteristics. In order to ensure the user a convenient way of working with the tool, a simple GUI was developed and added to RELOAD-M.

A logical and important further step is the parallelization. Two new features are also proposed that would make the tool more useful: the capability to interactively design the MYRRHA loading patterns manually by the user followed by an ordinary preliminary evaluation (without optimization), and the capability to tackle multicycle core-reload problems.

Chapter 6

Conclusion

This dissertation was set out to develop a core management tool for MYRRHA (Multi-purpose hYbrid Research Reactor for High-tech Applications) that is able to solve the loading pattern optimization problem (LPOP) or, in other words, to calculate an optimal core given a stockpile of fresh and burnt fuel assemblies. Considering the special character of the facility, which is conceived as a flexible fast-spectrum lead-bismuth-cooled irradiation facility capable of operating in both critical and subcritical mode, this dissertation has also sought to identify an appropriate optimality criterion and neutron-transport methods to be used for optimization.

This project was motivated by the need to fill the gap in the present reactor-core modeling capabilities of the MYRRHA team, by providing the team with a new analysis tool that is sufficiently fast, computationally inexpensive, and can be used for routine in-core fuel management calculations.

An extensive literature survey was conducted in order to examine the relevance of different in-core fuel management problems and analytical practices to MYRRHA, and to assess available optimization techniques used for tackling the loading pattern optimization problem. Several decisions were made based on this survey, taking into account the specifics of MYRRHA nuclear fuel management and reactor-core and fuel design:

- The MYRRHA loading pattern optimization problem was defined as a simple fuel reshuffling problem, in which one type of a fresh fuel assembly was used with one fuel-pin enrichment and without burnable poisons. This implies that no other subsidiary in-core fuel management optimization problem needs to be solved for MYRRHA, for instance, the fuel-lattice

design problem and some other problems typically solved for large power LWRs. Also, the influence of the orientation of the fuel assemblies in the loading-pattern on the reactor core characteristics was neglected, which is a standard and well-reasoned assumption commonly used for fast reactors. No type of prevalent refueling strategy was assumed for MYRRHA, meaning that every fuel assembly could take any position in the reactor core. This assumption is more typical for LWRs than for fast reactors, which are normally characterized by a zonal or scatter refueling strategy. It was adopted for MYRRHA to allow for a maximum variability within the search process and thus also for the best possible quality of the results.

- The maximum irradiation performance expressed in terms of the average fast-neutron fluence achieved in the experimental channels in the reactor core during one cycle was postulated as a reasonable optimization objective of the MYRRHA loading pattern optimization problem. This choice was made considering the primary purpose of the facility, which is material irradiation.

- Two basic constraints were identified for MYRRHA loading-pattern optimization, besides the given number of available fuel assemblies of different types that can be loaded into the reactor: The first constraint was the maximum cladding-temperature limit, which is the most severe thermal limit that ensures the fuel integrity during operation. The second constraint was the end-of-cycle (EOC) criticality condition, which is the main reactivity limit that ensures the required length of reactor operation. The second constraint applies only for the critical MYRRHA core configuration, which is the only configuration considered in this dissertation.

- An optimization approach based on metaheuristic optimization methods was adopted to solve the MYRRHA loading pattern optimization problem as the state-of-the-art approach delivering the highest-quality results in the area of in-core fuel management optimization. Among the myriad of available metaheuristic optimization methods, two iterative population-based methods well suited for solving the core-reload problem were selected for this work: a Genetic Algorithm (GA), a well-established method, and an Ant Colony Optimization (ACO) algorithm, a more recent method.

- The adopted optimization scheme relies on the use of black-box reactor analysis codes, which evaluate all candidate solutions (loading patterns) generated during the search, and return a demanded set of physics characteristics for each of them. Therefore, two codes (solvers) tailored for fast-reactor neutronics analysis were selected and used in this work:

DIF3D, a steady-state nodal diffusion/transport code, and REBUS, the DIF3D extension with a depletion capability. A simple numerical subchannel model was also implemented in the work for calculation of the maximum fuel-cladding temperature.

The remainder of this chapter is organized as follows: First, the original contributions of this work are summarized in a small section. Then, the main results and findings are reviewed in the next section, followed by two sections on applications and limitations of the work performed. And finally, some recommendations for future work are given in the last section.

Contributions

It is firmly believed that the originality and main added value of the performed research lies in the five following points:

- A versatile core management tool named RELOAD-M (REactor Loading Optimization And Design for MYRRHA) was developed to solve the MYRRHA single-cycle loading pattern optimization problem. It can integrate different optimization methods and reactor analysis codes. Various objective and constraint functions can be used for optimization, which is an important feature, considering the multipurpose character of MYRRHA.
- The loading-pattern optimization scheme usually used for large-scale LWRs for power generation was applied for the first time to a medium-scale fast-spectrum research reactor.
- A special type of optimization objective and optimization constraints were used in the loading pattern optimization problem that reflect the primary function of MYRRHA as a fast-spectrum lead-bismuth-cooled material testing reactor (MTR).
- The performances of two distinct metaheuristic optimization techniques applied to two core-reload problems were compared in terms of the quality of the obtained results, as well as in terms of the computational cost.
- Parametric studies were carried out in order to assess the impact different parameters have on the efficiency of both optimization methods when applied to a simple core-reload problem.

Findings

The main findings have been summarized at the end of Chapters 2, 3, and 4. This section synthesizes them in a few points:

- The results of the optimization study conducted in this dissertation showed that only a negligible improvement in the MYRRHA irradiation performance can be achieved by applying the adopted optimization scheme. In fact, the optimized value of the fast-neutron fluence in the experimental channels was only slightly more than one percent greater than in the case of some guessed solutions. Moreover, the latter solutions were usually characterized by smaller reactor powers and larger EOC effective multiplication factors, both characteristics speaking in favor of the guessed solutions. These observations put the whole concept of the MYRRHA loading-pattern optimization into question and call for a further investigation of alternative optimization objectives. It should be noted though that similar results were expected for a fast-spectrum system, which is generally less sensitive to changes in the reactor-core configuration than a thermal-spectrum system.
- The simple MYRRHA reactor-core diffusion model was found to be the preferred option for the loading-pattern optimization over the transport model mainly for the much shorter computing time. Both models delivered results of comparable accuracy.
- Out of the two optimization methods applied in this work, the elitist GA was the most efficient and consistent method and outperformed the non-elitist GA variant as well as the ACO algorithm. The elitist GA also needed the smallest average number of evaluations to find the optimum solution.
- The average time required for all of the methods to complete the optimization process was shorter than five hours on the reference computing platform. This is a very good result, especially if one considers that an overnight calculation was already regarded satisfactory, and that the computing time can be significantly further reduced, if the employed algorithms are parallelized.
- RELOAD-M can be already very useful for finding loading patterns complying with all imposed constraints. This is particularly true if multiple constraints are introduced into the problem.

Applications

Several applications are envisaged for the developed RELOAD-M core management tool and the adopted modeling framework:

- RELOAD-M is readily usable for performing various single-cycle MYRRHA reactor-core optimization studies with different optimization objectives and constraints. The same studies can also be done for other MYRRHA designs if the employed reactor-core physics models are adjusted appropriately. Ultimately, RELOAD-M could be used for loading-pattern optimization of other nuclear reactors like EFIT (European Facility for Industrial Transmutation). In this case, the maximum transmutation performance would probably be a more coherent optimization objective.
- RELOAD-M can also serve as a scoping tool for preliminary analysis of modified or new MYRRHA core designs. Currently, all proposals for a new MYRRHA core design or configuration are evaluated by a computationally very expensive high-fidelity Monte-Carlo code. Using RELOAD-M as a tool complementary to the Monte-Carlo code would significantly increase the number of different core designs that could be evaluated and, at the same time, tremendously decrease the required computing power. Only a few of the most promising designs could then be evaluated by a more precise method after all designs were sorted out and inferior ones eliminated by RELOAD-M. However, this would again require appropriate changes in the current MYRRHA core physics models.
- Another possible field of application of RELOAD-M is academic research. For instance, the tool could be used to benchmark different algorithms applied to a defined loading pattern optimization problem.

Limitations

The major shortcomings of the present work relate mostly to the limitations given by the adopted problem-solving approach and the capabilities of the developed optimization tool. They can be listed as follows:

- Only one loading pattern optimization problem was solved for MYRRHA, which aimed at maximizing the facility's irradiation capacity. It was concluded that the obtained improvement was minimal. Therefore, it would be desirable to consider some other optimization objectives to

see if other core characteristics are more prone to changes in the core arrangement.

- The current RELOAD-M version can only handle single-cycle problems. This means that medium- and long-term fuel-economy oriented in-core fuel management problems cannot be addressed presently by the tool.
- A third limitation has to do with the reactor-core models, which were employed in RELOAD-M for the calculation of the beginning-of-cycle (BOC) and EOC effective multiplication factors, and neutron-flux and power-density distributions. While already sufficient for carrying out basic loading-pattern optimization studies, these models do not supply enough information for more detailed evaluations of the analyzed loading patterns (e.g., various reactivity coefficients or shutdown-system performance).
- The employed reactor-core models describe only the critical MYRRHA reactor core.

Future Work

A number of future paths for research and development are suggested mainly to alleviate the limitations described in the preceding section and, also, to improve the usability and usefulness of the RELOAD-M core management tool:

- The first possible area of future research is closer investigation of alternative MYRRHA single-cycle optimization problems. This dissertation solved a research-oriented core-reload problem aimed at maximizing the irradiation performance of the machine. Other types of optimization objectives can be considered for MYRRHA. For instance, economy-oriented objectives such as maximizing the EOC multiplication factor or fuel discharge burnup, or production-oriented objectives such as maximizing molybdenum production or silicon-doping rates. It also would be interesting to revisit the original objective used in this work and its role as the most representative research-oriented objective.
- A special research area which has not been touched in this dissertation is multicycle optimization. Solving the multicycle loading pattern optimization problem may lead to fuel savings after several cycles, which would improve the economy of the operation of MYRRHA. Some type of multicycle analysis can already be performed with the current RELOAD-M version, if the loading patterns are optimized independently one-by-one for a series of successive fuel cycles. This scheme, however, does

not incorporate the multicycle coupling effect, which basically states that finding the best loading pattern for one fuel cycle may worsen the attainable quality of future loading patterns and, consequently, also the overall quality of the aggregate multicycle solution. In order to account for this effect, all RELOAD-M components need to be further developed, including the optimization methods, the reactor analysis models, and the tool itself.

- A lot of work can be done to improve the accuracy, scope, and computational cost of the reactor-physics models embedded in RELOAD-M. For example, a two-dimensional model of the MYRRHA core can be prepared and tested to enhance the computing time. Even more significant improvements in the computing time could be accomplished if some extremely fast surrogate reactor-physics models are used within the tool, such as artificial neural networks (ANNs). Of course, the models currently implemented within the tool are also open for improvement (fine-tuning, more accurate nuclear data treatment, etc.) A model of the subcritical MYRRHA core has to be prepared so the loading pattern optimization problem can also be solved for the subcritical MYRRHA core design. Analogously, a suitable reactor-physics model needs to be worked out if the multicycle optimization option should be included in the tool. In this case, it would be desirable to provide the tool with a module for an economic analysis of the MYRRHA fuel cycle.

- The effectivity and efficiency of the GA and ACO optimization algorithms, currently embodied in RELOAD-M, can be improved. For instance, it is recommended that both algorithms be combined with other local-search metaheuristics to strengthen their search qualities (hybridization). Also, both algorithms still need to be fine-tuned. The implementation of other advanced and (hopefully) better-performing optimization methods within the tool is also possible. The following proven methods are suggested for possible implementation: Tabu Search (TS), Particle Swarm Optimization (PSO), and some Pareto-based multiobjective methods like Non-dominated Sorting Genetic Algorithm II (NSGA-II).

- The work to be carried out to make RELOAD-M an improved and useful core management tool was described at the end of the preceding chapter. The main goals can be summarized as follows: parallelization of the optimization algorithms (which will also need some changes in the RELOAD-M source code) implementation of the multicycle optimization option and functionality extension with an interactive graphical interface for manual loading-pattern design.

- A large number of various optimization methods have been used for solving the loading pattern optimization problem. However, no benchmark has been set up yet for evaluating the performance of these methods when applied to the problem. Therefore, any contribution in this area would be original and very welcomed and appreciated by the research and engineering community.

Appendix A

Studies Survey

In the following table A.1, more than 300 studies on solving the loading pattern optimization problem are listed that have been collected by the author of this dissertation during his PhD. It should be viewed as a starting point for an interested reader, new to the area of loading pattern optimization, rather than an exhaustive overview with a complete set of publications on the topic, which it is not. A few notes on the table:

- The table is organized as follows (from left to right columns): bibliography references (see Bibliography for the full records), applied solution methods, considered objectives, considered constraints, and names of optimization tools, employed reactor analysis codes, and types of investigated reactors. Often, the name of a reactor code has the form "CODE1/CODE2", where CODE1 is a cross-section generation code and CODE2 is a full-core analysis code.
- The superscripts in the second column usually indicate special properties of the solved problem or applied method. They are explained in the footer of the table on page 213.
- The symbols $\nearrow$ and $\searrow$ mean maximization, respectively, minimization of objective functions, while the symbols $\uparrow$ and $\downarrow$ mean maximum, respectively, minimum constraint values.
- The abbreviations and symbols used in the table are explained in the corresponding chapters beginning on page xix and xxvi, respectively. The names of optimization tools and reactor codes were not included for the sake of brevity.

Table A.1: Survey of core reload optimization studies

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
1	Goertzel, 1956*	VOCT	↘ fuel cost				
2	Wall and Fenech, 1965*	DP	↘ fuel cost				
3	Fagan and Sesonske, 1969*	DP	↘ fuel cost				
4	Mélice, 1969*	VOCT	↗ k_{eff}				
5	Stover and Sesonske, 1969	ESES					BWR
6	Sauar, 1971*	LP	↘ fuel cost				
7	Suzuki and Kiyose, 1971a*	LP	↘ fuel cost			EOLEXD	BWR
8	Suzuki and Kiyose, 1971b*	VOCT	↗ B_{c}				
9	Wade and Terney, 1971*	VOCT	↗ B_{c}, ↘ PPF				
10	Goldschmidt, 1972*	VOCT	↘ fuel cost				
11	Hoshino, 1972*	heuristic/ AI	↗ B_{c}				
12	Naft and Sesonske, 1972*	DS	↘ PPF				
13	Howland et al., 1973						PWR
14	Mingle, 1973	LP, PT					
15	Stout and Robinson, 1973*	DP	↘ PPF				
16	Bell and Shapiro, 1974						
17	Chitkara and Weisman, 1974*	DS	↘ fuel cost				
18	Rieck et al., 1974*	DS	↘ fuel cost				
19	Rieck, 1974						PWR
20	Chen, 1975						LWR
21	Ciechanowiecz, 1975*	VOCT	↗ B_{c}				

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
22	Kubokawa and Kiyose, 1975	LP					
23	Mingle, 1975*	LP, PT	$\searrow$ fuel cost				
24	Motoda, Herczeg, et al., 1975*	LP, DS	$\searrow$ fuel cost				
25	Sekimizu, 1975						
26	Kobayashi et al., 1976	EHES	$\searrow$ PPF, $\searrow$ fresh FAs	$\downarrow k_{\text{eff}}$, $\uparrow$ RPPF, $\uparrow$ PPF, $\uparrow$ avg. B_{d}			LMFBR
27	Motoda and Yokomizo, 1976						BWR
28	Chen et al., 1977	heuristic/ LP	$\searrow$ RPPF				PWR
29	Kobayashi et al., 1977	MMIA	P flattening				FR
30	Motoda and Yokomizo, 1977*	LP, DS	$\searrow$ fuel cost				
31	Huang and S. Levine, 1978*	VOCT	$\nearrow B_{\text{c}}$				
32	Mingle, 1978		$\searrow$ PPF				
33	Huang, 1979		$\nearrow$ EOC k_{eff}				
34	B. I. Lin et al., 1979g,*	DS	$\nearrow k_{\text{eff}}$, $\nearrow$ fuel costs				
35	A. L. B. Ho, 1981	DS	$\searrow$ PPF			EPRI-NODE-P	PWR
36	D. Chang and S. H. Levine, 1982*	VOCT	$\searrow$ PPF				
37	L.-W. Ho and Rohach, 1982*	PT	$\searrow$ PPF				
38	A. L. B. Ho and Sensoke, 1982*	DS	$\nearrow k_{\text{eff}}$				
39	Terney and Williamson Jr., 1982*	VOCT	$\searrow$ PPF				
40	Colletti et al., 1983*	VOCT	$\nearrow k_{\text{eff}}$				
41	Izenson, 1983*	DP	$\searrow$ PPF				
42	Yeh et al., 1983				REFUEL		BWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
43	Ahn and S. H. Levine, 1984[*]	VOCT	$\nearrow B_c$	$\uparrow$ PPF		PSU-LEOPARD/DEAL-ENR	PWR
44	Y. C. Chang and Sesonske, 1984						PWR
45	Kyung-Eung, 1984	variational method					
46	Ahn and S. H. Levine, 1985	GPM				PSU-LEOPARD/DEAL-ENR	PWR
47	Sauer and Driscoll, 1985[*]	DS	$\nearrow B_c$				
48	Story and Grow, 1985[*]	heuristic/AI	$\nearrow k_{eff}$				
49	Chao, Hu, et al., 1986						
50	Downar and Y. J. Kim, 1986[g]						PWR
51	Hamasaki and Takeda, 1986	LP					
52	Hobson and Turinsky, 1986		$\nearrow$ BOC k_{eff}	$\uparrow$ PPF, $\uparrow B$			PWR
53	Y. J. Kim, 1986[g]		$\nearrow t_c$	$\uparrow$ PPF		SIMULATE	PWR
54	S. H. Levine, 1986[c,*]	SA					
55	Morita et al., 1986				LPOP		
56	Turinsky and Hobson, 1986[*]	PT	$\searrow$ fuel cost				
57	White et al., 1986[*]	PT	$\nearrow k_{eff}$				
58	Y. J. Kim et al., 1987[g,*]	LP, DS	$\nearrow B_c$				
59	Parks, 1987						GCR
60	Downar and Sesonske, 1988[c]						
61	Galperin and Nissan, 1988	heuristic					
62	Okafor and Aldemir, 1988	LP	$\nearrow$ BOC k_∞				PWR
63	Rothleder et al., 1988[*]	heuristic/AI	$\searrow$ PPF				

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
64	Galperin, Kimhi, et al., 1989	heuristic					
65	H. Kim, 1989[e,g]	LP	↗ t_c, ↗ EOC k_{eff}				LWR
66	H. C. Kim et al., 1989[e,g]	LP	↗ t_c			HUDDLE	BWR
67	Kropaczek, 1989	SA					
68	Stillman et al., 1989[*]	LP	↗ k_{eff}				
69	Suh, 1989[g]	LP	↘ fuel cycle cost				PWR
70	Kropaczek and Turinsky, 1990	PT					
71	Parks, 1990b	SA			AMETROP		
72	Parks, Turinsky, et al., 1990						PWR
73	Parks, 1990a[c]						
74	Suh and S. H. Levine, 1990						PWR
75	Zavaljsevski, 1990						PWR
76	Galperin and Kimhy, 1991	heuristic	↘ PPF		FUELCON		PWR
77	Kropaczek and Turinsky, 1991	SA					
78	Kropaczek, Parks, et al., 1991	SA					
79	S. H. Levine, 1991		↗ EOC k_{eff}				PWR
80	Stevens, Smith, and Downar, 1991						PWR
81	Tahara et al., 1991[g]		↘ RPPF				PWR
82	Parks and Lewins, 1992[c]						
83	Petrović and S. H. Levine, 1992[e]		↗ t_c	↑RPPF		MCRAC	PWR
84	Poon and Parks, 1992	SA, GA			FORMOSA, FORMOGA		PWR
85	Poon, 1992	GA					PWR
86	Shvedov and Goncharov, 1992	PVAM/ILP					FR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
87	Bai, 1993[e]						PWR
88	Bai et al., 1993				PFMP-SCO		
89	H. G. Kim et al., 1993[a]	heuristic, fuzzy rules	$\nearrow k_{\text{eff}}$	$\uparrow$ PPF			
90	Kropaczek, McElroy, et al., 1993[e]		$\nearrow B_{\text{d}}$, $\searrow x_{\text{f}}$		FORMOSA-P		PWR
91	S. H. Levine and Bai, 1993[e]		opt. EOC k_{eff}		PFMP-SCO, PFMP-MCSO	ADMARC/ PSUI-LEOPARD	PWR
92	Li, 1993	expert system					PWR
93	Poon and Parks, 1993[b]	GA	$\searrow$ PPF		FORMOSA		PWR
94	Rosset and Barral, 1993	trial-and-error	$\searrow$ PPF	$\uparrow$ hot pin P, $\downarrow$ EOC SDM, negative MTC, $\uparrow$ FA B			
95	Šmuc, Petrović, et al., 1993		$\searrow$ fueling cost	$\uparrow$ MTC, $\uparrow F_{\Delta H}$, $\uparrow B_{\text{d}}$	MOCALPS	PSU-LEOPARD/ MCYC1.5D	PWR
96	Stevens et al., 1993[e]	SA, DS	$\searrow$ RPPF	$\uparrow$ RPPF	X-IMAGE/ SIMAN	CASMO-3/ SIMULATE-3	PWR
97	Verhagen and Van der Schar, 1993	SA					LWR
98	Zhian and S. H. Levine, 1993[g]				AUTOLOAD		PWR
99	Kropaczek, Turinsky, et al., 1994				FORMOSA-P		PWR
100	S. H. Levine, Bai, and Zhian, 1994[e]		$\nearrow B_{\text{d}}$, $\nearrow$ EOC k_{eff}		PFMP-MCSO	PSUI-LEOPARD	PWR
101	S. H. Levine, John, et al., 1994[c]						
102	Li and S. H. Levine, 1994[g]		$\nearrow$ EOC k_{eff}		AUTOLOAD		PWR
103	Mahlers, 1994						PWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
104	Šmuc, Pevec, et al., 1994	SA					
105	E. Tanker and A. Tanker, 1994	GA	↗ EOC k_{eff}	t_{c}, PPF			PWR
106	DeChaine and Feltus, 1995b	GA	↗ EOC k_{eff}	↑ norm. P	CIGARO	CASMO-3/ SIMULATE-3	
107	DeChaine and Feltus, 1995a	GA	↗ BOC k_{eff}	↑ norm. P	CIGARO	CASMO-3/ SIMULATE-3	PWR
108	DeChaine, 1995	GA					
109	De Jong, 1995						
110	Galperin, 1995	heuristic	func. of k_{eff} and PPF	*see objectives*			PWR
111	van Geemert, 1995	CIA					
112	Koolwaaij, 1995	PT					
113	S. H. Levine, 1995[c]						
114	Mahlers, 1995	SA, LP					PWR
115	Maldonado and Turinsky, 1995	SA			FORMOSA-P		PWR
116	Maldonado, Turinsky, et al., 1995		↘ PPF, ↗ EOC k_{eff}, ↗ B_{d}, ↘ x_{f}	↑ PPF, t_{c} requirement, ↑ B_{d}, ↑ MTC, etc.	FORMOSA-P		PWR
117	Meneses, M. D. Machado, et al., 2009	PSORK	↘ norm. FA P	↑ FA rel. P		RECNOD	PWR
118	Parks, 1995	GA					PWR
119	Stevens, 1995						PWR
120	Stevens et al., 1995[e]	SA	↗ t_{c}, ↗ B_{d}, ↘ 2D pin peaking factor	↑ 2D pin peaking factor, ↓ t_{c}	X-IMAGE/ SIMAN	CASMO-3/ SIMULATE-3	PWR
121	Argaud, 1996	GIA	↘ PPF				PWR
122	DeChaine and Feltus, 1996	GA	↗ core lifetime	↑ PPF	CIGARO	CASMO-3/ SIMULATE-3	PWR
123	van Geemert, Quist, and E. J. Hoogenboom, 1996	CIA	↗ EOC k_{eff}	↑ BOC k_{eff}, ↓ SDM, ↑ PPF		COLD VENTURE	PWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
124	T. K. Kim and C. H. Kim, 1996	MIP					
125	Parks, 1996[d]	MOGA	$\searrow x_f$, $\nearrow B_d$, $\searrow$ RPPF	fixed B_c, fixed EOC B	FORMOSA-P	PANTHER	PWR
126	Yamamoto, 1996[e]	GA	$\searrow$ RPPF, $\nearrow t_c$	$\uparrow$ RPPF	GALLOP		PWR
127	Axmann, 1997[f]		func. of t_c, c_B, PPF, etc.	*see objectives*	RELOPAT, PRIMO	PRISM	PWR
128	Carter, 1997[b,c,*]	HC, SA, GDA, RRT, TS, GA, PBIL, MpA	$\searrow$ PPF		FORMOSA		PWR
129	van Geemert, J., et al., 1997	SA	$\nearrow$ EOC k_{eff}	$\uparrow$ BOC k_{eff}, $\downarrow$ SDM, $\uparrow$ PPF			RR
130	Keller and Turinsky, 1997				FORMOSA-P		PWR
131	T. K. Kim and C. H. Kim, 1997[a]	MIP/DLP, B&B	$\nearrow$ EOC k_{eff}			CASMO-3/ NEMSNAP	
132	Klerk et al., 1997[b]	MINLP	$\nearrow B_d$		GAMS/ DICOPT		PWR
133	De Lima, M. D. Machado, et al., 2007	AS					
134	Mahlers, 1997	MILP	$\nearrow$ local thermal neutron flux				RR
135	Parks, 1997	MOGA					PWR
136	Parks and Knight, 1997					PANTHER	
137	Schirru, Pereira, et al., 1997	GA					
138	Siegelmann et al., 1997[a]				FUELCON		
139	Verhagen, Van der Schaar, et al., 1997	SA	many functions	*see objectives*	ROSA	LWRWIMS/ LWRSIM	PWR
140	Yamamoto, 1997[b]	SA, DS, BE, GA, hybrids	$\nearrow t_c$	$\uparrow$ RPPF			PWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
141	Yamamoto and Kanda, 1997[b,e]	GA	↗ energy produced	↑ RPPF, ↑ FA B, ↓ MTC	OPAL, GALLOP, MCA	SHARP	PWR
142	Yamamoto, Noda, et al., 1997[a,e]	GA, BE	↘ RPPF, ↗ t_c, ↗ B_d	↑ FA , negative MTC, ↑ PPF	INSIGHT/ GALLOP/ MCA	SHARP	PWR
143	Zimmerman and Umbarger, 1997				X-IMAGE/ SIMAN		PWR
144	van Geemert, Quist, J. E. Hoogenboom, et al., 1998	CIA					RR
145	C. Lin, J.-I. Yang, et al., 1998	TS	↘ $F_{\Delta H}$				PWR
146	K.-J. Lin and C. Lin, 1998	expert system	↘ $F_{\Delta H}$				PWR
147	Nissan and Galperin, 1998				FUELCON		
148	Quist, Klerk, et al., 1998	MINLP	↗ EOC k_{eff}	↑ PPF			PWR
149	Yamamoto, 1998						PWR
150	Zhao et al., 1998	GA	↗ t_c	↑ PPF	FuelGen		PWR
151	Bolloni et al., 1999[f]	EA			MEPO, PRIMO, PARAT	PRISM, PRESTO-2	PWR
152	Bolloni, 1999[f]	EA					PWR
153	Chapot et al., 1999	GA	↘ PPF, ↗ t_c		GENESIS/ ALGER	ANC	PWR
154	J.-L. François and López, 1999[g]	GA	↗ t_c	↑ RPPF	SOPRAG	PRESTO-B	BWR
155	J.-L. François, Martín del Campo, et al., 1999					PRESTO-B	BWR
156	van Geemert, 1999	heuristic, PT					
157	Ben Hmaida et al., 1999	TS					
158	Kato et al., 1999[e]	LP	↘ PPF				FBR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
159	H. C. Lee et al., 1999[f]	SA					
160	M. D. Machado, 1999						
161	Moore et al., 1999	SA	$\nearrow$ EOC k_{eff}, $\searrow$ PPF, $\nearrow$ CPR, $\nearrow$ B_{d}, $\searrow$ reload cost	thermal and fuel related limits and energy production	FORMOSA-B		BWR
162	Parks and Suppapitnarm, 1999	MOSA					PWR
163	Quist, van Geemert, J. E. Hoogenboom, Illés, C. Roos, et al., 1999	MINLP	$\nearrow$ EOC k_{eff}	$\uparrow$ PPF			PWR
164	Quist, van Geemert, J. E. Hoogenboom, Illés, Klerk, et al., 1999	MINLP	$\nearrow$ EOC k_{eff}	$\uparrow$ PPF			PWR
165	Quist, van Geemert, J. E. Hoogenboom, De Klerk, et al., 1999[b]	MINLP, SA, PI	$\nearrow$ EOC k_{eff}	$\uparrow$ PPF	DICOPT		PWR
166	Toshinsky et al., 1999	GA				SLAROM/ CITATION	LMFBR
167	Turinsky, 1999[c]	GA, SA, TS					
168	Turinsky and Parks, 1999[c]	GA, SA					
169	Van de Velde et al., 1999	EA	t_{c}, BOC c_{B}, $\nearrow$ PPF, etc.	*see objectives*	PRIMO, PARAT	PRISM, PRESTO-2	PWR
170	Chapot, 2000	GA					
171	Karve and Turinsky, 2000	SA			FORMOSA-B	SIMULATE-3/ MICROBURN-B2	BWR
172	Shatilla et al., 2000[d]				ALPS		
173	Wantz, 2000[b]	EA/GA	$\searrow$ BOC c_{B}, $\nearrow$ t_{c}, $\nearrow$ local P, $\searrow$ FA number	*see objectives*	PRIMO M, PRIMO B/ RELOGS		

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
174	Yamamoto and Hashimoto, 2000	TPSA	$\nearrow t_c$	$\uparrow$ PPF, $\uparrow B_d$, $\uparrow$ MTC, $\uparrow$ rad. P tilt	INSIGHT/ GALLOP	SHARP	PWR
175	Hongchun, 2001	GA	$\searrow$ PPF, $\nearrow$ EOC c_B	$\uparrow$ PPF, $\downarrow t_c$		TPFAP/ NGFMARC	PWR
176	Jagawa et al., 2001	LPM, TS	$\nearrow$ EOC k_{eff}	thermal margins, $\downarrow$ SDM	FINELOAD-3	SIMULATE-3	BWR
177	Jang et al., 2001[a]	SA					PWR
178	Karve and Turinsky, 2001	SA		critical flow constr., $\downarrow$ SDM, thermal margin	FORMOSA-B	SIMULATE-3/ MICROBURN-B2	BWR
179	Karve, Keller, et al., 2001						
180	H. C. Lee et al., 2001[f]	SA					PWR
181	Quist, K. Roos, et al., 2001	MINLP	$\nearrow$ EOC k_{eff}	$\uparrow$ PPF			PWR
182	Yamamoto, 2001a[c]						LWR
183	Yamamoto, 2001b[e,f]	GA, GDA, SA, TPSA, DS, BE, et al.			INSIGHT/ GALLOP/ MCA	SIMULATE3/ SHARP	PWR
184	Ziver, Carter, Pain, C. R. E. d. Oliveira, and Goddard, 2001a				GAOPT		
185	Ziver, Carter, Pain, C. R. E. d. Oliveira, and Goddard, 2001b				GAOPT		
186	Allaire and Castro, 2002	homogen. method	$\nearrow k_{eff}$	$\uparrow$ PPF			PWR
187	Chao, Si, et al., 2002[a]	B&B, B3PEC	$\nearrow t_c$	$\uparrow$ PPF	LP-FUN	ANC	
188	Guler, 2002						PWR
189	Kobayashi and Aiyoshi, 2002[g]	GA	func. of EOC k_{eff} and Haling P peak				BWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
190	L. Machado and Schirru, 2002[b]	Ant-Q	$\searrow$ RPPF			RECNOD	PWR
191	Mahlers, 2002	SA, LP					
192	Sadighi et al., 2002b[b]	SA	$\searrow$ FPF				PWR
193	Sadighi et al., 2002a[b]	SA	$\searrow$ FPF				PWR
194	Yamamoto and Hashimoto, 2002[f]	DGA					
195	Ziver, Carter, Pain, C. R. E. d. Oliveira, Goddard, and Overton, 2002[e]		$\searrow$ RPPF, $\nearrow$ station-specific profit function, $\nearrow B_{\mathrm{d}}$	$\downarrow$ irradiation limit, $\uparrow$ number of rad. shuffles, $\uparrow$ RPPF, avg. CR insertion limit, etc.	GAOPT	PANTHER	AGR
196	Boroushaki et al., 2003	GA, fuzzy NLP	$\nearrow k_{\mathrm{eff}}$ and $\searrow$ PPF			CITATION	PWR
197	Daubert et al., 2003	SA					BWR
198	Erdoğan and Geçkinli, 2003[a]	GA	func. of k_{eff} and PPF		Xcore		PWR
199	Kobayashi and Aiyoshi, 2003b[g]	GA					BWR
200	Kobayashi and Aiyoshi, 2003a	GA					BWR
201	Lam and al., 2003[f]	SA			LP-FUN		
202	Oyarzun, 2003						BWR
203	Stevens and Rempe, 2003[e]	SA, DS	$\nearrow t_{\mathrm{c}}$	$\uparrow$ MTC, $\uparrow F_Q$, $\uparrow F_{\Delta H}$	X-IMAGE/ SIMAN	CASMO-4/ SIMULATE-3	PWR, BWR
204	Verhagen and Wakker, 2003	SA	many functions	*see objectives*	ROSA		PWR
205	Verhagen, Wakker, and Bloois, 2003	SA	$\searrow$ reloading time		ROSA		PWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
206	Martín del Campo, J. L. François, et al., 2004	GA	func. of energy produced, MLHGR, XMPGR, MRNP, MCPR, MFAB, HER, SDM	*see objectives*		CM-PRESTO	BWR
207	Castillo, Alonso, et al., 2004	TS	func. of energy produced, MRNP, RPPF, XLHGR, XMPGR, XMCPR, SDM	*see objectives*	OTSS	CM-PRESTO-B	BWR
208	Guler et al., 2004[g]		↗EOC k_{eff}	↑ norm. P	SCAM-W, CIGARO	Moby-Dick	PWR
209	Ortiz and Requena, 2004a[a]	GA	↗ BOC or EOC k_{eff}	↑ RPPF, MCPR, MLHGR	RECOPIA	CM-PRESTO	BWR
210	Ortiz and Requena, 2004b[a]		↗ BOC k_{eff}	↑ RPPF, MCPR, MLHGR			BWR
211	Yamamoto, Sugimura, et al., 2004[e]		func. of t_{c}, ↗ rel. FA P, avg. B_{d}	*see objectives*			PWR
212	Ziver, Pain, et al., 2004[a]	GA	↘ RPPF, ↗ avg. B_{d}, ↗ per-day-profit func.		GAOPT	PANTHER	AGR
213	M. D. Machado, 2005	PBIL					
214	Meneses, 2005						
215	Sato, 2005[a]	B&B, MILP	↗ EOC c_{B}, ↘ PPF	*see objectives*	Pearls™		
216	Turinsky, 2005[c]						
217	Turinsky, Keller, et al., 2005[c]						LWR
218	Alim, 2006	GA	↗ norm. P, ↗ EOC c_{B}	↑ PPP	GARCO	Moby-Dick, CASMO-4, SIMULATE-3	PWR
219	Do et al., 2006a	GA	↗ B_{d}, ↘ max. channel P, ↘ max. change in ZCU	↑ B_{d}, ↑ channel P, ↑ bundle P, ↑ PPF, ↓ ZCU	AUTOREFUEL		PHWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
220	Do et al., 2006b	EA					PHWR
221	Jiang et al., 2006[a]	GA, EDA	↗ k_{eff}			EVENT	RR
222	Mazrou and Hamadouche, 2006[a]	SA	func. of k_{eff} and PPF	*see objectives*			RR
223	Meneses and Schirru, 2006	PSORK					
224	Schirru, Lima, et al., 2006[f]	PBIL, ACS	↗ t_{c}, ↘ MARP	↑ FA rel. P		RECNOD	PWR
225	Castillo, Ortiz, et al., 2007[g]	TS	func. of k_{eff}, MFLPD, MPGR, MFLCPR, SDM, HER	*see objectives*	QuinalliBT	CM-PRESTO 3D	BWR
226	Chambon et al., 2007	AGM	↘ fueling cost	↓ k_{eff}, ↑ PPF	OPTEX	DONJON	PHWR
227	Do and Nguyen, 2007	GA	↗ k_{eff}	↑ PPF		CITATION	RR
228	Ortiz, Castillo, et al., 2007[a]	GA, ACO	↗ BOC k_{eff}	↓ BOC SDM, range of HER, MCPR, MLHGR	OCOTH, RENOR	CM-PRESTO	BWR
229	Alim, Ivanov, and S. H. Levine, 2008a	GA	↗ norm. P, ↗ EOC c_{B}	↑ PPP	GARCO	CASMO-4/ SIMULATE-3	PWR
230	Alim, Ivanov, Yilmaz, et al., 2008[g]	GA	↗ norm. P, ↗ EOC c_{B}	↑ PPP	GARCO	CASMO-4/ SIMULATE-3	PWR
231	Alim, Ivanov, and S. H. Levine, 2008b[g]	GA	↗ norm. P, ↗ EOC c_{B}	↑ PPP	GARCO	CASMO-4/ SIMULATE-3	PWR
232	Caldas and Schirru, 2008	PBIL, FPBIL	↗ critical c_{B}				PWR
233	Fadaei and Setayeshi, 2008[a]	SA	↘ PPF		LONSA	WIMS-4D/ CITATION	PWR
234	Hoareau, 2008	ACO, MMAS	↗ t_{c}, ↘ vessel fluence	↑ PPF		COCCINELLE	PWR
235	Iqbal et al., 2008		↗ k_{eff}	peak-to-average and flux-ratio constr.	BFMTR	WIMS-4D/ CITATION	MTR
236	De Lima, Schirru, et al., 2008[f]	ACS	func. of RPPF and c_{B}	*see objectives*			PWR
237	Alim, Yilmaz, et al., 2009	GA	↗ EOC k_{eff}		GARCO	SIMULATE-3	PWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
238	Babazadeh et al., 2009[d]	VEPSO	func. of k_{eff} and PPF	*see objectives*		WIMS-D5B/ CITATION	PWR
239	Martín del Campo, Palomera-Pérez, et al., 2009	GA	func. of energy produced, MLHGR, XMPGR, MRNP, MCPR, MFAB, HER, SDM	*see objectives*		CM-PRESTO	BWR
240	Fadaei and Setayeshi, 2009	CA	$\nearrow$ k_{eff} and $\searrow$ RPPF			WIMS-4D/ CITATION	PWR
241	Fadaei, Setayeshi, and Kia, 2009	CA	$\searrow$ RPPF			WIMS-4D/ CITATION	PWR
242	Hays, 2009[f]	SA	$\nearrow$ EOC k_{eff}, $\nearrow$ CPR, MFLPD, $\nearrow$ B_{d}	$\uparrow$ B_{d}, thermal margins, constr. on allowable coolant flows, $\uparrow$ and $\downarrow$ HER, $\downarrow$ SDM	FORMOSA-B		BWR
243	Hedayat et al., 2009[a,d]	NSGA-II	$\nearrow$ thermal ϕ in a flux trap, $\nearrow$ life time	$\uparrow$ PPF, CR weight	Core Pattern Calculator 1	WIMS-D5/ CITVAP V3.2	RR
244	Jiang, 2009[a]	GA, EDA	$\nearrow$ k_{eff}			EVENT	RR
245	Khoshahval, 2009						PWR
246	Mishra et al., 2009	EDA	$\nearrow$ k_{eff}	$\uparrow$ bundle/channel P, channel outlet T, and permitted SDM			PHWR
247	T. K. Park et al., 2009[d]	MOSA	$\nearrow$ t_{c}, $\searrow$ RPPF	$\uparrow$ MTC, $\uparrow$ rad. pin peaking factor, etc.		UNCARDS	PWR
248	Sacco et al., 2009[b]	DE, DERL	$\searrow$ ϕ or PPF			HAMMER	PWR
249	Waintraub et al., 2009[f]	PPSO	$\nearrow$ EOC c_{B}	$\uparrow$ FA rel. P		RECNOD	PWR
250	Wang and C. Lin, 2009[g]	RAS	func. of t_{c}, SDM, and MFLCPR	*see objectives*		SIMULATE-3	BWR
251	Fadaei, N. M. Moghaddam, et al., 2010	CA	$\searrow$ PPF			WIMS-4D/ CITATION	PWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
252	Ishida and Sekimoto, 2010b	DP	↗ MA transmut. rate	limits on B_d, k_{eff}, and PPF			ADS
253	Ishida and Sekimoto, 2010a	DP	↗ MA transmut. rate	limits on B_d, k_{eff}, and PPF			ADS
254	Keyvani et al., 2010		↘ PPF			MCNP-4C	RR
255	Khoshahval, Zolfaghari, et al., 2010[b]	PSO	↗ P flattening	↑ PPF	CRCPSO	WIMS-D4/ CITATON LDI-2	PWR
256	Meneses, Rancoita, et al., 2010	CBS	↗ EOC c_B	↑ FA rel. P		RECNOD	PWR
257	Meneses, Gambardella, et al., 2010	RNAH, PSORK, RS	↗ c_B	↑ PPF		RECNOD	PWR
258	Meneses, Lima, et al., 2010[c]	GA, PBIL, ACO, PSO					
259	Shaukat et al., 2010		↗ t_c, ↗ thermal ϕ in the flux trap				RR
260	Silva and Schirru, 2010[b]	QPBIL	↗ c_B	↑ PPF		RECNOD	PWR
261	Turinsky, 2010[c]						
262	Esquivel-Estrada et al., 2011[g]	ACS, AS, BWAS, MMAS	func. of k_{eff}, MFLCPR, MFLPD, MAPRAT, SDM	*see objectives*	Azcaxalli	CM-PRESTO	BWR
263	J.-L. François, Guzmáin, et al., 2011	GA	↗ energy produced	↑ PPF, ↑ MLHGR, ↑ XMPGR, ↑ MRNP, ↑ MFAB, ↓ SDM, ↓ and ↑ ΔHER		HELIOS/ TABGEN/ PRESTO	BWR
264	Gong et al., 2011[d]	IBA, GA	↘ , ↗ t_c	*see objectives*	IBALPO, GALPO	SMART	PWR
265	Khoshahval, H. Minuchehr, et al., 2011[b]	PSO, GA	↗ P flattening	↑ PPF		WIMS-D4/ CITATON LDI-2	PWR
266	Knott and Oyarzun, 2011				MARLA	SIMULATE-3	BWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
267	I. M. S. d. Oliveira and Schirru, 2011	ABCRK	↗ c_B	↑ PPF		RECNOD	PWR
268	Ortiz-Servin et al., 2011[a]		func. of EOC k_{eff}, SDM, MFLCPR, MFLPD, MPGR	*see objectives*	OCONN		BWR
269	Safarzadeh, Zolfaghari, Norouzi, et al., 2011[b]	ABC	↘ PPF				PWR
270	Silva, Schirru, and Lima, 2011[b]	QACO _Alpha	func. of c_B and PPF			RECNOD	PWR
271	Yilmaz et al., 2011				ePrometheus		BWR
272	Abbassi et al., 2012	ACO	↗ P flattening		FEMPT-PSO		PWR
273	Dai et al., 2012[d]	NSGA-II	↘ total fuel potential in k_{eff}, ↘ PPF				PWR
274	Haghighattalab et al., 2012[b]	BIMCMC	↗ k_{eff}	↑ PPF		WIMS-4D/ CITATION	PWR
275	Hosseini and Vosoughi, 2012	PT	↘ PPF			WIMS/ CITATION	PWR
276	Jaluvka et al., 2012	GA	↗ BOC k_{eff}			SCALE 6/ DIF3D10.0	MTR, FR
277	Kalcheva and Koonen, 2012	heuristic	↗ BOC k_{eff}, ↗ fuel savings	↓ SDM		MCNPX 2.7.0	MTR, RR
278	C. Lin and B.-F. Lin, 2012	RAS, MMAS, Ant-Q	func. of t_c, MTC, and $F_{\Delta H}$	*see objectives*		SIMULATE-3	PWR
279	Liu and Cai, 2012	IPPSO	↗ k_{eff}	↑ PPF			PWR
280	Nicolau et al., 2012[b]	QAE	↗ c_B	↑ PPF		RECNOD	PWR
281	Pazirandeh and Tayefi, 2012[a]		↘ PPF			WIMS-4D/ CITATION	PWR
282	Rahmani et al., 2012	WFM	↗ k_{eff}, ↘ RPPF	*see objectives*		WIMSD5-B/ CITATION LDI-2/ COBRA-EN	PWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
283	Yadav and Gupta, 2012	PSO	$\searrow$ PPF	$\uparrow$ PPF	PRISHA		PWR
284	Aghaie et al., 2013	HS, IHS	$\nearrow k_{\text{eff}}$	$\uparrow$ PPF		PARCS/ COBRA-EN	PWR
285	J.-L. François, Ortiz-Servin, et al., 2013[a,b,c]	GA, TS, AS, MMAS, BWAS, ACS	$\nearrow$ energy produced	$\uparrow$ PPF, $\uparrow$ MLHGR, $\uparrow$ XMPGR, $\uparrow$ MRNP, $\uparrow$ MFAB, $\downarrow$ SDM, $\downarrow$ and $\uparrow$ ΔHER		HELIOS/ TABGEN/ PRESTO	BWR
286	Jaluvka et al., 2013b	GA, ACO	$\nearrow \phi$	$\downarrow$ BOC k_{eff}		SCALE 6/ DIF3D10.0	MTR, FR
287	Jamalipour, Sayareh, et al., 2013	QPSO, QPSO-DM	func. of k_{eff} and PPF	*see objectives*		WIMS/ CITATION	PWR
288	Jamalipour, Gharib, et al., 2013	QPSO, QPSO-DM	$\nearrow$ FA P flattening	$\uparrow$ PPF		WIMS/ CITATION	PWR
289	Karahroudi et al., 2013	GA	$\nearrow k_{\text{eff}}$	$\uparrow$ RPPF		WIMS-4D/ CITATION	PWR
290	Nazari et al., 2013	IHS	$\nearrow k_{\text{eff}}$	$\uparrow$ PPF		CITATION	PWR
291	Norouzi et al., 2013	PICGA	$\nearrow k_{\text{eff}}$	$\uparrow$ PPF		WIMS-4D/ PARCS	RR, PWR
292	Pelykh et al., 2013						PWR
293	Poursalehi, Zolfaghari, and A. Minuchehr, 2013c	HS, GHS, SGHS	$\nearrow$ FA P flattening	$\uparrow$ PPF	SGHSNE		PWR
294	Poursalehi, Zolfaghari, and A. Minuchehr, 2013a	DHS	FA P flattening, $\searrow$ PPF	*see objectives*	DHSNEP-2D		PWR
295	Poursalehi, Zolfaghari, A. Minuchehr, and H. K. Moghaddam, 2013	CFA	func. of k_{eff} and RPPF	*see objectives*	CFANEC		PWR
296	Poursalehi, Zolfaghari, and A. Minuchehr, 2013b	DFA	$\nearrow k_{\text{eff}}$, $\searrow$ PPF		DFANEC		PWR
297	Poursalehi, Zolfaghari, A. Minuchehr, and Valavi, 2013	SGHS	FA P flattening, $\searrow$ PPF	*see objectives*	SGHSNE		PWR

#	Reference	Method(s)	Objective(s)	Constraint(s)	Opt. tool	R. code	R. type
298	Schlünz et al., 2013[e]	HS	$\searrow$ EOC discharge mass, $\nearrow$ thermal ϕ, $\nearrow$ ^{99}Mo production and Si doping capacity	$\uparrow$ PPF, $\downarrow$ SDM, $\downarrow$ CBW	OSCAR-4		RR
299	Tayefi and Pazirandeh, 2013[a]		$\nearrow$ k_{eff}, $\searrow$ PPF			WIMS-4D/ CITATION	PWR
300	Thakur et al., 2013				CARSH		PHWR
301	Kashi et al., 2014	BA	func. of k_{eff} and PPF	*see objectives*	BANEC		PWR
302	Safarzadeh, Zolfaghari, Zangian, et al., 2014	ABC	$\nearrow$ BOC k_{eff}	$\uparrow$ PPF		WIMS-D5/ CITATON	PWR
303	Silva and Schirru, 2014[b]	SQPBIL	func. of c_{B} and PPF			RECNOD	PWR
304	Hill and Parks, 2015[b]	TS					BWR

[a] Application of artificial neural networks (ANNs).
[b] Benchmark or performance comparison of different optimization methods.
[c] Review or overview of different LPOP solving methods.
[d] Multiobjective optimization.
[e] Multicycle optimization.
[f] Application of parallel computing techniques.
[g] Application of the Haling principle (HP).
[*] Taken from Driscoll et al., 1990.

Appendix B

Thermal-Hydraulics Module

The thermal-hydraulics analysis performed during MYRRHA loading pattern optimization aims at calculation of the maximum fuel cladding temperature T_{clad} for every loading pattern evaluated. The value of T_{clad} is obtained based on a simple analytical subchannel model developed by Castelliti (2012).

Input Parameters

The model is supported by four parameters on its input; two parameters coming from the results of the neutronics analysis—the radial and axial power peaking factors P_{r} and P_{z}, and two parameters characteristic for the analyzed core—the relative power fraction p_{f} and the total number of fuel assemblies N_{FA}. The last two parameters are required to allow a user to use the model for MYRRHA core designs different from the reference one. In the following, the reference core design will be assumed with $N_{\mathrm{FA}} = 69$ fuel assemblies, power output of $100\,\mathrm{MW}$ ($p_{\mathrm{f}} = 1.0$), coolant mass flow rate $m_{\mathrm{core}} \approx 4.3 \times 10^3\,\mathrm{kg\,s^{-1}}$, and inlet coolant temperature $T_{\mathrm{in}} = 270\,^{\circ}\mathrm{C}$.

Hot Fuel Assembly Analysis

An important thermal-hydraulics characteristic is the maximum linear power

$$q_{\mathrm{max}} = P_{\mathrm{r}}\, P_{\mathrm{z}}\, \bar{q} \tag{B.1}$$

calculated as the product of the power peaking factors and the average linear power

$$\bar{q} = \frac{P}{L_{\mathrm{act}}\, N_{\mathrm{FA}}\, (N_{\mathrm{pin}} - 1)} \ , \tag{B.2}$$

where $L_{\mathrm{act}} = 60\,\mathrm{cm}$ is the active length and $N_{\mathrm{pin}} = 127$ is the number of pins in a single fuel assembly.

Hot Subchannel Analysis

The total number of subchannels N_{SCH} in the adopted subchannel model is 258. Out of them $N_{\mathrm{inn}} = 216$ are internal subchannels, $N_{\mathrm{edg}} = 36$ are edge subchannels, and $N_{\mathrm{cor}} = 6$ are corner subchannels. The speed distribution among the fuel assembly subchannels $(v_{\mathrm{inn}}, v_{\mathrm{edg}}, v_{\mathrm{cor}})^{\mathrm{T}}$ is obtained as the non-trivial solution of the system of equations describing the Novendstern independent channels model (Castelliti, 2012):

$$\begin{pmatrix} d_{\mathrm{h\,edg}}^{\frac{5}{7}} & -d_{\mathrm{h\,inn}}^{\frac{5}{7}} & 0 \\ 0 & d_{\mathrm{h\,cor}}^{\frac{5}{7}} & -d_{\mathrm{h\,edg}}^{\frac{5}{7}} \\ N_{\mathrm{inn}}\, A_{\mathrm{inn}} & N_{\mathrm{edg}}\, A_{\mathrm{edg}} & N_{\mathrm{cor}}\, A_{\mathrm{cor}} \end{pmatrix} \begin{pmatrix} v_{\mathrm{inn}} \\ v_{\mathrm{edg}} \\ v_{\mathrm{cor}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ v_{\mathrm{avg}}\, A_{\mathrm{tot}} \end{pmatrix} \ . \tag{B.3}$$

In the system B.3, $d_{\mathrm{h}\,x}$ and A_x are hydraulic diameters, respectively, areas of inner, edge, and corner subchannels, and $A_{\mathrm{tot}} = \sum_x A_x$.

With the subchannel speeds known, first, the inner subchannel mass flow rate m_{inn} and, then, the subchannel mass flow rate peaking factor f_{SCH} can be calculated that is used below in the $T_{\mathrm{hotSCH}}(z)$ formula for calculating T_{clad}:

$$m_{\mathrm{inn}} = v_{\mathrm{inn}}\, A_{\mathrm{inn}}\, \rho_{\mathrm{LBE}} \ , \tag{B.4}$$

$$f_{\mathrm{SCH}} = \frac{m_{\mathrm{FA}}}{m_{\mathrm{inn}}\, N_{\mathrm{SCH}}} \ . \tag{B.5}$$

ρ_{LBE} in Eq. (B.4) is the density of the coolant at its average temperature and m_{FA} is the average fuel assembly mass flow rate, $m_{\mathrm{FA}} = m_{\mathrm{core}}\, N_{\mathrm{FA}}^{-1}$.

According to the Castelliti's model, T_{clad} is calculated as the temperature on the outer surface of the cladding in the hottest subchannel in the reactor $T_{\mathrm{hotSCH}}(z)$ in its hottest position in the axial direction Z_{hotSCH}:

$$T_{\mathrm{clad}} = T_{\mathrm{hotSCH}}(Z_{\mathrm{hotSCH}}) \ . \tag{B.6}$$

The function $T_{\mathrm{hotSCH}}(z)$ of axial position z is defined as

$$T_{\mathrm{hotSCH}}(z) = T_{\mathrm{in}} + P_{\mathrm{z}}\, P_{\mathrm{r}}\, q \left(\frac{L_{\mathrm{ext}}\, f_{\mathrm{SCH}}}{\pi\, m_{\mathrm{pin}}\, c_{\mathrm{p\,LBE}}} \left(\sin\left(\frac{\pi\, z}{L_{\mathrm{ext}}} \right) + \sin\left(\frac{\pi\, L_{\mathrm{act}}}{2\, L_{\mathrm{ext}}} \right) \right) + \frac{\ln\left(\frac{D+2\,\delta_{\mathrm{ox}}}{D} \right)}{2\,\pi\, k_{\mathrm{ox}}} + \frac{1}{\pi\, D\, h} \cos\left(\frac{\pi\, z}{L_{\mathrm{ext}}} \right) \right) , \quad \text{(B.7)}$$

where

L_{ext} is the extrapolated length satisfying the relation

$$L_{\mathrm{ext}} = \frac{\pi}{2} \frac{L_{\mathrm{act}}}{P_{\mathrm{z}}} \frac{1}{\sin\left(\frac{\pi}{2} \frac{L_{\mathrm{act}}}{L_{\mathrm{ext}}} \right)} , \quad \text{(B.8)}$$

m_{pin} is the pin average mass flow rate, $m_{\mathrm{pin}} = m_{\mathrm{FA}}\, N_{\mathrm{pin}}^{-1}$,
$c_{\mathrm{p\,LBE}}$ is the LBE heat capacity at the average hot subchannel temperature,
D is the pin outer diameter,
δ_{ox} is the thickness of the oxide layer on the cladding surface,
k_{ox} is the oxide layer thermal conductivity, and
h is the coolant heat transfer coefficient.

The position Z_{hotSCH} is find as a solution of the equation

$$\frac{\mathrm{d}T_{\mathrm{hotSCH}}(z)}{\mathrm{d}z} = 0 . \quad \text{(B.9)}$$

The heat transfer coefficient h used in Eq. (B.7) is calculated assuming the average temperature on the outlet of the hot fuel assembly as follows:

$$h = \frac{Nu\, k_{\mathrm{LBE}}}{d_{\mathrm{h\,tot}}} . \quad \text{(B.10)}$$

Here, k_{LBE} is the coolant thermal conductivity, $d_{\mathrm{h\,tot}} = \sum_x d_{\mathrm{h}\,x}$, and the Nusselt number

$$Nu = 7.55\, \frac{P_{\mathrm{t}}}{D} - 20 \left(\frac{P_{\mathrm{t}}}{D} \right)^{-13} + \frac{3.67}{90} \left(\frac{P_{\mathrm{t}}}{D} \right)^{-2} Pe^{0.56+0.19\, \frac{P_{\mathrm{t}}}{D}} ; \quad \text{(B.11)}$$

P_{t} is the modified pin pitch and Pe is the Péclet number, the product of the Reynolds number Re and the Prandtl number Pr, $Pe = Re\, Pr$.

Appendix C

Material Compositions and Mixtures

The values presented in Tables C.1–C.8 describe nuclide compositions of the essential materials used in the MYRRHA-FASTEF reactor core. Table C.9 then shows mixing ratios for the material mixtures directly used in the MYRRHA-FASTEF models described in Chapter 3 (see Figure 3.4). All presented data are based on the information taken from Sarotto, 2011. The ^{4}He atom density values (Table C.8) were calculated applying the ideal gas law.

Table C.1: LBE composition (at 270, 360, and 410 °C)

	Composition			
Temp. (°C)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.	At. density ($b^{-1}\,cm^{-1}$)
270	Pb	1.3418×10^{-2}	^{209}Bi	1.6600×10^{-2}
360	Pb	1.3266×10^{-2}	^{209}Bi	1.6400×10^{-2}
410	Pb	1.3185×10^{-2}	^{209}Bi	1.6300×10^{-2}

[a] Natural isotopic abundance assumed.
Source: Sarotto, 2011.

Table C.2: 15-15Ti SS composition (at 20 °C)

Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)
C	3.5900×10^{-4}	^{31}P	6.9600×10^{-5}	^{184}W	2.3900×10^{-6}
Cr	1.3358×10^{-2}	^{14}N	5.1100×10^{-5}	^{186}W	2.2000×10^{-6}
Ni	1.2651×10^{-2}	^{15}N	1.7500×10^{-7}	^{93}Nb	7.7300×10^{-6}
^{55}Mn	1.3100×10^{-3}	S	2.2357×10^{-5}	^{181}Ta	3.9500×10^{-6}
Mo	7.4870×10^{-4}	^{27}Al	2.6600×10^{-5}	Cu	2.2620×10^{-5}
Ti	3.9980×10^{-4}	Zr	1.5749×10^{-5}	^{59}Co	2.4400×10^{-5}
Si	1.4477×10^{-3}	V	2.8200×10^{-5}	Ca	3.5805×10^{-5}
^{10}B	5.7100×10^{-6}	^{182}W	2.1000×10^{-6}	Fe	5.6003×10^{-2}
^{11}B	2.0900×10^{-5}	^{183}W	1.1200×10^{-6}		

[a] Natural isotopic abundance is assumed where no mass number is specified.
Source: Sarotto, 2011.

Table C.3: T91 FMS composition (at 20 °C)

Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)
C	3.8900×10^{-4}	Si	5.8230×10^{-4}	S	7.2845×10^{-6}
Cr	7.8690×10^{-3}	^{93}Nb	4.0200×10^{-5}	Cu	7.3600×10^{-5}
Ni	1.5948×10^{-4}	^{31}P	3.0200×10^{-5}	V	1.9700×10^{-4}
^{55}Mn	3.8300×10^{-4}	^{14}N	1.6600×10^{-4}	^{27}Al	6.9300×10^{-5}
Mo	4.6270×10^{-4}	^{15}N	5.7000×10^{-7}	Fe	7.4232×10^{-2}

[a] Natural isotopic abundance is assumed where no mass number is specified.
Source: Sarotto, 2011.

Table C.4: 316L SS composition (at 20 °C)

Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)
C	1.1900×10^{-4}	Si	1.6964×10^{-3}	^{15}N	1.2800×10^{-6}
Cr	1.6093×10^{-2}	^{10}B	1.7100×10^{-6}	S	2.2355×10^{-5}
Ni	9.3546×10^{-3}	^{11}B	6.2500×10^{-6}	Cu	7.5200×10^{-4}
^{55}Mn	1.7400×10^{-3}	^{31}P	4.6300×10^{-5}	^{59}Co	1.6200×10^{-4}
Mo	1.1210×10^{-3}	^{14}N	3.7400×10^{-4}	Fe	5.5020×10^{-2}

[a] Natural isotopic abundance is assumed where no mass number is specified.
Source: Sarotto, 2011.

Table C.5: AlMgSi1 composition (at 20 °C)

Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)
Mg	6.6990×10^{-4}	Ti	3.3990×10^{-5}	Fe	1.4524×10^{-4}
^{27}Al	5.7900×10^{-2}	Cr	7.8220×10^{-5}	Cu	2.5620×10^{-5}
Si	6.3730×10^{-4}	^{55}Mn	2.2200×10^{-4}	Zn	4.9700×10^{-5}

[a] Natural isotopic abundance is assumed where no mass number is specified.
Source: Sarotto, 2011.

Table C.6: B_4C composition (at 20 °C)

Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.	At. density ($b^{-1}\,cm^{-1}$)	Nucl.	At. density ($b^{-1}\,cm^{-1}$)
^{10}B	8.9300×10^{-2}	^{11}B	9.9200×10^{-3}	C	2.4800×10^{-2}

[a] Natural isotopic abundance assumed.
Source: Sarotto, 2011.

Table C.7: YZrO composition (at 20 °C)

Nucl.	At. density ($b^{-1}\,cm^{-1}$)	Nucl.[a]	At. density ($b^{-1}\,cm^{-1}$)	Nucl.	At. density ($b^{-1}\,cm^{-1}$)
^{16}O	5.8100×10^{-2}	Zr	2.7808×10^{-2}	^{89}Y	1.6300×10^{-3}

[a] Natural isotopic abundance assumed.
Source: Sarotto, 2011.

Table C.8: ^{4}He atom density (at 1 and 5 bar)

Pressure (bar)	At. density ($b^{-1}\,cm^{-1}$)	Pressure (bar)	At. density ($b^{-1}\,cm^{-1}$)
1	2.4724×10^{-5}	5	1.2362×10^{-4}

Table C.9: Material mixtures

Mixture	Material volume fraction (%)									
	MOX_i	AlMgSi1	T91	316L	15-15Ti	B_4C	YZrO	LBE	^{4}He (1 bar)	^{4}He (5 bar)
B4C_CR			6.64		8.64	40.40		36.39	7.93	
B4C_SR			22.09	1.93	1.99	19.13		54.37	0.48	
BARREL1				98.30				1.70		
BARREL2				61.50				38.50		
BARREL3				53.20				46.80		
BARREL4				28.10				71.90		
BEAM_TB			11.21					29.78		
FLOW_DEV1			14.61					85.39		
FLOW_DEV2			21.61					78.39		
$FUEL_i$	30.74		7.29		14.81			44.22		2.93
GRID_CR			6.64	3.53	2.82			87.00		
GRID_PIN			7.29		48.48			44.22		
GRID_SR			22.09	40.69				37.22		
GRIDIN_SR			22.09	57.88				20.03		
HE_IPS				17.91				13.60	68.49	
LBE								100.00		
LBE_WRAP			7.29					92.71		
LBE_WRAPX			14.29					85.71		
NOZZLE			32.41					67.59		
PLEN_IPS		51.05		13.02				13.60	22.33	
PLENA_CR			6.64	3.05	8.64			36.39	45.27	
PLENAINF			7.29	7.30	14.81			44.22		26.37

Mixture	Material volume fraction (%)									
	MOX_i	AlMgSi1	T91	316L	15-15Ti	B_4C	YZrO	LBE	^{4}He (1 bar)	^{4}He (5 bar)
PLENASUP			7.29		18.18			44.22		30.30
ROLLER_CR			6.64	10.76				82.59		
ROLLER_SR			22.09	10.76				67.15		
SAMPLE_IPS		55.08	0.91	13.02				13.60	17.39	
SPAL_NOZZLE			25.87					74.13		
SPAL_TAR			11.14					14.00		
TOPPLATE			65.98					34.02		
TUBE_SR			22.09					77.91		
YZRO_CR			6.64		8.64		40.40	36.39	7.93	
YZRO_INSU			7.29		14.81		30.98	44.22		2.69
YZRO_IPS		30.29		13.02			25.67	13.60	17.42	
YZRO_SR			22.09	1.93	1.99		19.13	54.37	0.48	

Based on information from Sarotto, 2011.

Appendix D

Lumped Fission Products

Table D.1 contains 100 fission products present in the MYRRHA-FASTEF fuel with the largest macroscopic absorption cross sections $\Sigma_a = N\,\sigma_a$. These fission products were homogenized and used in the models in Chapter 3 as a single artificial fission product LFP (see Table 3.5). They are ordered in decreasing order and represent 99.9 % of the total Σ_a of all fission products and 99.0 % of all fission product atoms in the fuel. The values were obtained based on one-group cross sections σ_a and atomic densities N calculated by ALEPH using the JEFF-3.1.1 library (Santamarina et al., 2009) for the last fuel batch at the beginning of equilibrium cycle one step before unloading (Stankovskiy, 2013). The sum of the fission product yields is equal to 2.0.

Table D.1: Lumped fission products

#	FP	Yield	#	FP	Yield
1	^{105}Pd	5.39×10^{-2}	51	^{85}Rb	4.60×10^{-3}
2	^{101}Ru	6.18×10^{-2}	52	^{110}Pd	7.60×10^{-3}
3	^{99}Tc	5.88×10^{-2}	53	^{83}Kr	2.83×10^{-3}
4	^{103}Rh	5.98×10^{-2}	54	^{134}Cs	7.62×10^{-4}
5	^{107}Pd	3.25×10^{-2}	55	^{156}Gd	1.22×10^{-3}
6	^{133}Cs	6.80×10^{-2}	56	^{154}Eu	2.20×10^{-4}
7	^{149}Sm	1.23×10^{-2}	57	^{154}Sm	2.80×10^{-3}
8	^{151}Sm	7.22×10^{-3}	58	^{144}Ce	2.39×10^{-2}
9	^{97}Mo	5.17×10^{-2}	59	^{113}Cd	1.06×10^{-3}
10	^{135}Cs	7.30×10^{-2}	60	^{159}Tb	2.92×10^{-4}
11	^{147}Pm	1.74×10^{-2}	61	^{140}Ce	5.39×10^{-2}

#	FP	Yield	#	FP	Yield
12	^{145}Nd	3.08×10^{-2}	62	^{91}Y	4.12×10^{-3}
13	^{143}Nd	4.38×10^{-2}	63	^{138}Ba	6.12×10^{-2}
14	^{109}Ag	1.59×10^{-2}	64	^{125}Sb	1.04×10^{-3}
15	^{95}Mo	3.30×10^{-2}	65	^{89}Y	1.57×10^{-2}
16	^{102}Ru	6.35×10^{-2}	66	^{112}Cd	1.64×10^{-3}
17	^{131}Xe	3.66×10^{-2}	67	^{150}Sm	7.06×10^{-4}
18	^{104}Ru	6.19×10^{-2}	68	^{128}Te	8.25×10^{-3}
19	^{153}Eu	3.74×10^{-3}	69	^{130}Te	2.62×10^{-2}
20	^{98}Mo	5.65×10^{-2}	70	^{155}Gd	1.30×10^{-4}
21	^{100}Mo	6.62×10^{-2}	71	^{148m}Pm	4.64×10^{-5}
22	^{108}Pd	2.28×10^{-2}	72	^{129m}Te	3.31×10^{-4}
23	^{141}Pr	4.77×10^{-2}	73	^{90}Sr	2.07×10^{-2}
24	^{93}Zr	3.93×10^{-2}	74	^{104}Pd	8.24×10^{-4}
25	^{152}Sm	6.46×10^{-3}	75	^{87}Rb	1.05×10^{-2}
26	^{106}Pd	1.45×10^{-2}	76	^{127m}Te	2.49×10^{-4}
27	^{129}I	1.29×10^{-2}	77	^{148}Sm	7.92×10^{-4}
28	^{103}Ru	5.99×10^{-3}	78	^{123}Sb	7.09×10^{-4}
29	^{147}Sm	2.52×10^{-3}	79	^{121}Sb	4.58×10^{-4}
30	^{106}Ru	2.95×10^{-2}	80	^{161}Dy	1.03×10^{-4}
31	^{111}Cd	3.64×10^{-3}	81	^{100}Ru	1.03×10^{-3}
32	^{127}I	4.18×10^{-3}	82	^{115}In	5.38×10^{-4}
33	^{146}Nd	2.65×10^{-2}	83	^{84}Kr	4.89×10^{-3}
34	^{148}Nd	1.74×10^{-2}	84	^{158}Gd	5.48×10^{-4}
35	^{132}Xe	5.11×10^{-2}	85	^{81}Br	1.35×10^{-3}
36	^{95}Nb	5.90×10^{-3}	86	^{79}Se	4.15×10^{-4}
37	^{139}La	5.86×10^{-2}	87	^{85}Kr	1.29×10^{-3}
38	^{155}Eu	1.56×10^{-3}	88	^{147}Nd	1.41×10^{-4}
39	^{91}Zr	2.17×10^{-2}	89	^{117}Sn	4.79×10^{-4}
40	^{92}Zr	3.10×10^{-2}	90	^{114}Cd	7.74×10^{-4}
41	^{134}Xe	7.16×10^{-2}	91	^{136}Xe	6.99×10^{-2}
42	^{150}Nd	1.03×10^{-2}	92	^{136}Ba	1.31×10^{-3}
43	^{94}Zr	4.24×10^{-2}	93	^{110}Cd	3.51×10^{-4}
44	^{96}Zr	4.94×10^{-2}	94	^{151}Eu	2.96×10^{-5}
45	^{95}Zr	8.49×10^{-3}	95	^{119}Sn	3.51×10^{-4}
46	^{142}Ce	4.96×10^{-2}	96	^{88}Sr	1.40×10^{-2}
47	^{144}Nd	1.50×10^{-2}	97	^{125}Te	1.42×10^{-4}
48	^{137}Cs	6.37×10^{-2}	98	^{137}Ba	8.91×10^{-4}
49	^{157}Gd	8.69×10^{-4}	99	^{82}Se	2.06×10^{-3}
50	^{141}Ce	3.46×10^{-3}	100	^{162}Dy	7.04×10^{-5}

Values calculated based on ALEPH results (Stankovskiy, 2013).

Notation

The following notation is based on the ISO 31/XI standard summarized in the paper by Beccari (1997). According to the most important typesetting rules, variables and physical constants are always slanted, including vectors and Greek letters; numerical constants and mathematical operators are set upright; and vectors are in bold, including Greek letters. Symbol indexes (subscripts and superscripts) are set in italics when they represent physical quantities or mathematical variables, otherwise they are set as a normal text (upright). Different indexes of the same variable, no matter what is their type, are separated by a small space, whereas indexes of matrix elements are separated by a comma. Also important is the notation for multiplication, which is mostly denoted by a small space between the two operands, but can be omitted or substituted by the symbol "$\times$" in some special cases. Note also the different styles of the Greek letter sigma used for the summation symbol and for cross sections.

Object	Example	C.[a]	Note
Variables	$x, y, i, E, N, \omega, Pr, Re$	✓	
	Ω	×	
Constants	k, C, α	✓	
	$\mathrm{e} \approx 2.718, \pi \approx 3.142$	✓	numerical constants
	$\mathrm{i}^2 = -1$	✓	imaginary unit
	$e \approx 1.602 \times 10^{-19}\,\mathrm{C}$	✓	physical constants
	$\pi = \pi\,\mathrm{rad}$	✓	physical constants
	$\pm\infty$		infinity
Vectors	$\boldsymbol{x}, \boldsymbol{r}, \boldsymbol{o}, \tilde{\boldsymbol{o}}, \boldsymbol{\Omega}, \boldsymbol{C}$	×	
Functions	$f(x), \phi(\boldsymbol{r}, E), \psi(\boldsymbol{r}, E, \boldsymbol{\Omega})$		
Matrices	$\mathbb{A}, \mathbb{D}, \mathbb{F}, \mathbb{M}$		
Operators	$\mathrm{A}\psi = \mathrm{B}\psi + C$		
	$\sin\omega, \lim_{x\to 0} f(x)$	✓	
	$\int_a^b x \,\mathrm{d}x$	✓	

Object	Example	C.[a]	Note
	$\sum_{i=1}^{n} \frac{1}{i}$		
Sets	$\mathcal{F}, \mathcal{P}, \mathcal{S}$		
Number sets	$\mathbf{N}, \mathbf{Z}, \mathbf{Q}, \mathbf{C}, \mathbf{R}_{(0)}^{(\pm)}$		
Multiplication	$x\,y, x\,f(x), k\omega, \alpha f(x)$		
	$c = a \times b$		
	$y = x\,(x+1)$		
	$\mathbb{M} = \mathbb{A}\,\mathbb{F}$		matrices
Indexing	$x_{\mathrm{desc}}, x_{\mathrm{opt}}^{\lim}$	✓	descriptive
	$x_i, x_{i\,j}, a_{i,j} \equiv \mathbb{A}$	✓	variables
	$x_{\mathrm{desc}\,i\,j}$	✓	combined
Miscellaneous	$\mathrm{e}^{\,\mathrm{i}\pi} + 1 = 0$	✓	
	$\boldsymbol{\Omega} = (\varphi, \vartheta)$	×	
	$\sin k\omega \equiv \sin(k\omega)$	✓	for ≤ 2 operands
	$\frac{\mathrm{d}\,f(x)}{\mathrm{d}x}, \frac{\partial^2 f(x)}{\partial x^2}$		
	$\int_{4\pi} f(\boldsymbol{\Omega})\,\mathrm{d}^2\Omega$	×	
	$(\mathrm{A}\psi)(\boldsymbol{x}) = (\mathrm{B}\psi)(\boldsymbol{x}) + C$		
	$\mathbb{A}\boldsymbol{\phi} = \mathbb{F}\boldsymbol{\phi} + \boldsymbol{C}$	×	
	$\mathcal{P} = \{f(x) \vert x \in \mathcal{F}\}$		
	$s = \vert\mathcal{S}\vert$		cardinality
Physics	$\Sigma, \Sigma_{\mathrm{f}}, \sigma, \sigma_{\mathrm{f}}, \sigma_{\mathrm{f}\,g}$		cross sections

[a] Compliant with Beccari, 1997.

Bibliography

Abbassi, M., A. Zolfaghari, A. Minuchehr, and F. Khoshahval (2012). "A P_N-Based Approach Along PSO Scheme for PWR Core Reloading Patterns Optimization". In: *Nuclear Engineering and Design* 248, pp. 206–215 (pp. 31, 211).

Abd Elmoatty, M. S., M. S. Nagy, M. N. Aly, and M. K. Shaat (2011). "An Integrated Expert System for Optimum In Core Fuel Management". In: *Nuclear Engineering and Design* 241.9, pp. 3707–3718 (p. 27).

Abrahams, D. (2014). *Boost.Python*. URL: `www.boost.org/doc/libs/1_57_0/libs/python/doc/index.html` (p. 181).

Aghaie, M., T. Nazari, A. Zolfaghari, A. Minuchehr, and A. Shirani (2013). "Investigation of PWR Core Optimization using Harmony Search Algorithms". In: *Annals of Nuclear Energy* 57, pp. 1–15 (pp. 32, 212).

Ahn, D. H. and S. H. Levine (1984). "Direct Placement of Fuel Assemblies using the Gradient Projection Method". In: *Transactions of the American Nuclear Society* 46, pp. 123–125 (p. 198).

Ahn, D. H. and S. H. Levine (1985). "Automatic Optimized Reload and Depletion Method for a Pressurized Water Reactor". In: *Nuclear Technology* 71.3, pp. 535–547 (p. 198).

Aït Abderrahim, H., P. Kupschus, E. Malambu, P. Benoit, K. Van Tiechelen, B. Arien, F. Vermeersch, P. D'hondt, Y. Jongen, S. Ternier, and D. Vandeplassche (2001). "MYRRHA: A Multipurpose Accelerator Driven System for Research & Development". In: *Nuclear Instruments and Methods in Physics Research* A.463, pp. 487–494 (pp. 43, 44).

Aït Abderrahim, H., A. Al Mazouzi, B. Arien, P. Baeten, D. De Bruyn, D. Maes, E. Malambu, P. Schuurmans, M. Schyns, V. Sobolev, G. Van den Eynde, and D. Vandeplassche (2012). *MYRRHA Project—Multi-purpose hYbrid Research Reactor for High-tech Applications at Mol (Belgium): MYRRHA Technical Description*. Technical Report. Confidential. Mol, Belgium: SCK•CEN (pp. 6, 43–48, 51, 52).

Aït Abderrahim, H. and et al. (2012). "MYRRHA—Multi-Purpose Fast Spectrum Research Reactor". In: *Energy Conversion and Management* 63, pp. 4–10 (p. 44).

Aït Abderrahim, H. and P. Baeten (2012). "MYRRHA: A Multi-purpose hYbrid Research Reactor for High-tech Applications". In: *Proceedings of the International Conference PHYSOR 2012—Advances in Reactor Physics—Linking Research, Industry, and Education*. Knoxville, Tennessee, USA: American Nuclear Society (ANS) (pp. 43, 44).

Aït Abderrahim, H., D. De Bruyn, and et al. (2007). *MYRRHA Project—Technical Description*. Technical Report ANS/HAA/DDB/3900.B043000/85/07-17bis. Mol, Belgium: SCK•CEN (p. 44).

Aït Abderrahim, H., D. De Bruyn, and M. Giot (2007). "From MYRRHA to XT-ADS—Development of Pb-Bi Cooled ADS and Perspective of Implementation in Europe". In: *Proceedings of the International Conference Nuclear Energy for New Europe*. 901. Portorož, Slovenia (p. 44).

Aït Abderrahim, H., D. De Bruyn, G. Van den Eynde, and S. Michiels (2014). "Transmutation of High-Level Nuclear Waste by Means of Accelerator Driven System". In: *Wiley Interdisciplinary Reviews: Energy and Environment* 3.1, pp. 60–69 (pp. 6, 45, 47).

Aizawa, N., F. Kubo, and T. Iwasaki (2013). "Comparison of Different Neutronics Analysis Technique for Accelerator-Driven System". In: *Annals of Nuclear Energy* 60, pp. 368–373 (p. 40).

Alim, F. (2006). "Heuristic Rules Embedded Genetic Algorithm For In-Core Fuel Management Optimization". PhD Dissertation. Pennsylvania, USA: Pennsylvania State University (pp. 104, 109, 114, 207).

Alim, F., K. Ivanov, and S. H. Levine (2008a). "New Genetic Algorithms (GA) to Optimize PWR Reactors: Part I: Loading Pattern and Burnable Poison Placement Optimization Techniques for PWRs". In: *Annals of Nuclear Energy* 35.1, pp. 93–112 (pp. 109, 114, 208).

Alim, F., K. Ivanov, and S. H. Levine (2008b). "New Genetic Algorithms (GA) to Optimize PWR Reactors: Part III: The Haling Power Depletion Method for In-Core Fuel Management Analysis". In: *Annals of Nuclear Energy* 35.1, pp. 121–131 (p. 208).

Alim, F., K. Ivanov, S. Yilmaz, and S. H. Levine (2008). "New Genetic Algorithms (GA) to Optimize PWR Reactors: Part II: Simultaneous Optimization of Loading Pattern and Burnable Poison Placement for the TMI-1 Reactor". In: *Annals of Nuclear Energy* 35.1, pp. 113–120 (p. 208).

Alim, F., S. Yilmaz, K. Ivanov, and S. H. Levine (2009). "Genetic Algroithms to Automatically Optimize PWR Fuel Management Calculations". In: *Proceeding of the 4th Topical Meeting on Advances in Nuclear Fuel Management 2009 (ANFM IV)*. Vol. 1. Hilton Head Island, South Carolina, USA: American Nuclear Society (ANS) (pp. 31, 101, 208).

Allaire, G. and C. Castro (2002). "Optimization of Nuclear Fuel Reloading by the Homogenization Method". English. In: *Structural and Multidisciplinary Optimization* 24 (1), pp. 11–22 (pp. 28, 205).

ANL (2014a). *DIF3D*. Argonne National Laboratory (ANL). URL: `http://www.ne.anl.gov/codes/dif3d/` (p. 65).

ANL (2014b). *ETOE-2*. Argonne National Laboratory (ANL). URL: `http://www.ne.anl.gov/codes/etoe-2/` (p. 65).

ANL (2014c). *MC^2-2*. Argonne National Laboratory (ANL). URL: `http://www.ne.anl.gov/codes/mc2-2/` (p. 64).

ANL (2014d). *REBUS-3*. Argonne National Laboratory (ANL). URL: `http://www.ne.anl.gov/codes/rebus/` (p. 67).

Applegate, D. L., R. E. Bixby, V. Chvátal, and W. J. Cook (2007). *The Traveling Salesman Problem: A Computational Study*. 2nd. Princeton Series in Applied Mathematics. Princeton University Press (p. 103).

Argaud, J.-P. (1996). *A Loading Pattern Optimization Method for Nuclear Fuel Management*. Technical Report 97NB00004. Clamart Cedex, France: Electricté de France, Département Mécanique et Modéles Numerigués (p. 201).

Artioli, C., X. Chen, F. Gabrielli, G. Glinatsis, P. Liu, W. Maschek, C. Petrovich, A. Rineiski, M. Sarotto, and M. Schikorr (2008). "Minor Actinide Transmutation in ADS: The EFIT Core Design". In: *Proceeding of the International Conference on the Physics of Reactors: "Nuclear Power: A Sustainable Resource" (PHYSOR 2008)*. Casino-Kursaal Conference Center, Interlaken, Switzerland (p. 47).

Axmann, J. K. (1997). "Parallel Adaptive Evolutionary Algorithms for PWR Reload Pattern Optimization". In: *Nuclear Technology* 119.3, pp. 293–313, 276–291 (p. 202).

Babazadeh, D., M. Boroushaki, and C. Lucas (2009). "Optimization of Fuel Core Loading Pattern Design in a VVER Nuclear Power Reactors using Particle Swarm Optimization (PSO)". In: *Annals of Nuclear Energy* 36.7, pp. 923–930 (pp. 31, 209).

Baeten, P., R. Fernandez, D. De Bruyn, G. Van den Eynde, E. Malambu, P. Benoit, and H. Aït Abderrahim (2011). "MYRRHA, a Flexible Fast Spectrum Irradiation Facility".

In: *Transactions of the European Research Reactor Conference RRFM 2011*. Rome, Italy: European Nuclear Society (ENS), pp. 16–20 (pp. 6, 44).

Bai, D., S. H. Levine, J. D. Luoma, and M. Mahgerefteh (1993). "A Single-Cycle PWR Automatic Optimum Reload Code, PFMP-SCO". In: *Transactions of the American Nuclear Society* 69, pp. 455–456 (p. 200).

Bai, D. (1993). "A Comprehensive Set of Techniques to Automate the PWR Loading Pattern Design for Single and Multiple Cycles in an Optimum Manner". PhD Dissertation. Philadelphia, Pennsylvania, USA: Pennsylvania State University (p. 200).

Baluja, S. and S. Davies (1998). "Fast Probabilistic Modeling for Combinatorial Optimization". In: *AAAI '98/IAAI '98: Proceedings of the fifteenth national/tenth conference on Artificial Intelligence/Innovative applications of Artificial Intelligence*. Madison, Wisconsin, USA: American Association for Artificial Intelligence, pp. 469–476 (p. 31).

Bateman, H. (1910). "The Solution of a System of Differential Equations Occuring in the Theory of Radio-Active Transfromations". In: *Proceedings of the Cambridge Philosophical Society, Mathematical, and Physical Sciences*, pp. 423–427 (p. 41).

Bean, J. C. (1994). "Genetic Algorithms and Random Keys for Sequencing and Optimization". In: *ORSA Journal on Computing* 6.2, pp. 154–160 (p. 104).

Beccari, C. (1997). "Typesetting Mathematics for Science and Technology According to ISO 31/XI". In: *TUGboat* 18.1, pp. 39–48 (pp. 227, 228).

Bell, D. E. and J. F. Shapiro (1974). *Nuclear Reactor Refueling Optimization* (p. 196).

Ben Hmaida, I. A., J. N. Carter, C. R. E. De Oliveira, A. J. H. Goddard, and G. T. Parks (1999). "Nuclear In-Core Fuel Management Optimization using the Tabu Search Method". In: *Mathematics & Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*. Madrid, Spain (pp. 31, 203).

Birattari, M. (2009). *Tuning Metaheuristics: A Machine Learning Perspective*. Ed. by J. Kacprzyk. 2nd. Vol. 197. Studies in Computational Intelligence. Springer (p. 129).

Blickle, T. and L. Thiele (1995). "A Mathematical Analysis of Tournament Selection". In: *Proceedings of the Sixth International Conference on Genetic Algorithms*. Ed. by L. Eshelman. San Francisco, California, USA: Morgan Kaufmann Publishers Inc., pp. 9–16 (p. 110).

Blum, C. (2005). "Ant Colony Optimization: Introduction and Recent Trends". In: *Physics of Life Reviews* 2, pp. 353–373 (pp. 30, 120, 121, 127).

Bolloni, H.-W. (1999). *Parallel Adaptive Evolutionary Algorithms for PWR Reload Pattern Optimization System*. Technical Report, Siemens AG Nuclear Power Generation Methods and Codes. Siemens (p. 203).

Bolloni, H.-W., J. K. Axmann, and H. Finnemann (1999). "Performance Analysis of a Parallelized Loading Pattern Optimization Problem on Different Multiprocessor Architectures". In: *Mathematical Methods and Supercomputing in Nuclear Applications*. Madrid, Spain (pp. 184, 203).

Boroushaki, M., M. B. Ghofrani, and C. Lucas (2003). "Optimal Fuel Loading Pattern Design in PWR Nuclear Power Reactors using Genetic Algorithms and Fuzzy Nonlinear Programming". In: *Journal of Intelligent & Fuzzy Systems* 14, pp. 85–93 (pp. 101, 109, 112, 206).

Boussaïd, I., J. Lepagnot, and P. Siarry (2013). "A Survey on Optimization Metaheuristics". In: *Information Sciences* 237. Prediction, Control and Diagnosis using Advanced Neural Computations, pp. 82–117 (pp. 30, 120).

SCK•CEN (2011). *BR2, Research Rector with Multiple Applications*. Mol, Belgium (p. 42).

Bradfute, J. L., Y. A. Shatilla, and B. J. Johansen (1997). "Recent Developments in Westinghouse Automated Fuel Management Code, ALPS". In: *Advances in Nuclear Fuel Management II (ANFM 1997)*. American Nuclear Society. Myrtle Beach, South Carolina, USA, pp. 8–49 (p. 174).

Bubelis, E. and M. Schikorr (2012). *FASTEF Safety Analysis—Critical and Subcritical Mode (Part 1: Critical FASTEF Safety Analysis)*. Technical Report INR-01/12, NUKLEAR 3434. Confidential. Karlsruhe, Germany: Karlsruhe Institute of Technology (KIT) (p. 6).

Caldas, G. H. F. and R. Schirru (2008). "Parameterless Evolutionary Algorithm applied to the Nuclear Reload Problem". In: *Annals of Nuclear Energy* 35.4, pp. 583–590 (p. 208).

Canonical Ltd. (2014). *Ubuntu*. Canonical Ltd. URL: `www.ubuntu.com/` (p. 176).

Cantú-Paz, E. (1998). *A Survey of Parallel Genetic Algorithms*. Study (p. 184).

Carter, J. N. (1997). "Advances in Nuclear Science and Technology". In: vol. 25. Advances in Nuclear Science & Technology. Springer. Chap. Genetic Algorithms for Incore Fuel Management and Other Recent Developments in Optimization, pp. 113–154 (pp. 101, 103, 108–111, 115, 202).

Castelliti, D. (2012). *Private Communication* (pp. 215, 216).

Castillo, A., G. Alonso, L. B. Morales, C. Martín del Campo, J. L. François, and E. del Valle (2004). "BWR Fuel Reloads Design using a Tabu Search Technique". In: *Annals of Nuclear Energy* 31.2, pp. 151–161 (pp. 31, 207).

Castillo, A., J. J. Ortiz, J. L. Montes, and R. Perusquía (2007). "Fuel Loading and Control Rod Patterns Optimization in a BWR using Tabu Search". In: *Annals of Nuclear Energy* 34.3, pp. 207–212 (p. 208).

Chadwick, M. B., M. Herman, P. Obložinský, M. E. Dunn, Y. Danon, A. C. Kahler, D. L. Smith, B. Pritychenko, G. Arbanas, R. Arcilla, R. Brewer, D. A. Brown, R. Capote, A. D. Carlson, Y. S. Cho, H. Derrien, K. Guber, G. M. Hale, S. Hoblit, S. Holloway, T. D. Johnson, T. Kawano, B. C. Kiedrowski, H. Kim, S. Kunieda, N. M. Larson, L. Leal, J. P. Lestone, R. C. Little, E. A. McCutchan, R. A. MacFarlane, M. MacInnes, C. M. Mattoon, R. D. McKnight, S. F. Mughabghab, G. P. A. Nobre, G. Palmiotti, A. Palumbo, M. T. Pigni, V. G. Pronyaev, R. O. Sayer, A. A. Sonzogni, N. C. Summers, P. Talou, I. Thompson, A. Trkov, R. L. Vogt, S. C. van der Marck, A. Wallner, M. C. White, D. Wiarda, and P. G. Young (2011). "ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data". In: *Nuclear Data Sheets* 112.12. Special Issue on ENDF/B-VII.1 Library, pp. 2887–2996 (pp. 78, 84).

Chambon, R., E. Varin, and D. Rozon (2007). "CANDU Fuel Management Optimization using Alternative Gradient Methods". In: *Annals of Nuclear Energy* 34.12, pp. 1002–1013 (p. 208).

Chang, D. and S. H. Levine (1982). "Optimal Placement and Poison Control Depletion of Individual Assemblies using the Gradient Project Method". In: *Transcriptions of the American Nuclear Society*. Vol. 43. American Nuclear Society (ANS), p. 163 (p. 197).

Chang, Y. C. and A. Sesonske (1984). "Optimization and Analysis of Low-Leakage Core Management for Pressurized Water Reactors". In: *Nuclear Technology* 65.2, pp. 292–304 (p. 198).

Chao, Y.-A., S. Si, H. Q. Lam, D. Sato, F. D. Popa, and D. C. Little (2002). "Loading Pattern Search by Branching and Bounding Batch Patterns enumerated under Constraints". In: *Proceeding of the International Conference on the New Frontiers of Nuclear Technology: Reactor Physics, Safety and High-Performance Computing (PHYSOR 2002)*. Sheraton Walker Hill Hotel, Seoul, Korea (p. 205).

Chao, Y.-A., C.-W. Hu, and C.-A. Suo (1986). "A Theory of Fuel Management via Backward Diffusion Calculation". In: *Nuclear Science and Engineering* 93.1, pp. 78–87 (p. 198).

Chapot, J. L. C. (2000). "Otimização Automática de Recargas de Reatores a Água Pressurizada Utilizando Algoritmos Genéticos". PhD Dissertation. Brazil: COPPE/UFRJ (p. 204).

Chapot, J. L. C., F. R. Silva, and R. Schirru (1999). "A New Approach to the Use of Genetic Algorithms to Solve the Pressurized Water Reactor's Fuel Management Optimization Problem". In: *Annals of Nuclear Energy* 26, pp. 641–655 (pp. 101, 112, 203).

Chen, Y. F., J. O. Mingle, and N. D. Eckhoff (1977). "Optimal Power Profile Fuel Management". In: *Annals of Nuclear Energy* 4, pp. 407–415 (p. 197).

Chen, Y. F. (1975). "Optimal Refueling Pattern Design for Light Water Reactors". PhD Dissertation (p. 196).

Chitkara, K. and J. Weisman (1974). "Equilibrium Approach to Optimal In-Core Fuel Management for Pressurized Water Reactors". In: *Nuclear Technology* 24.1, pp. 33–49 (p. 196).

Ciechanowiecz, W. (1975). "A Multilevel Approach to Nuclear Fuel Burnup Optimization". In: *Nuclear Science and Engineering* 57, p. 39 (p. 196).

Coello Coello, C. A. (2002). "Theoretical and Numerical Constraint-Handling Techniques used with Evolutionary Algorithms: A Survey of the State of the Art". In: *Computer Methods in Applied Mechanics and Engineering* 191.11-12, pp. 1245–1287 (pp. 151, 153–155).

Colletti, J. P., S. H. Levine, and J. B. Lewi (1983). "Iterative Solution to the Optimal Poison Management Problem in Pressurized Water Reactors". In: *Nuclear Technology* 63.3, pp. 415–425 (p. 197).

Dai, C., X. Wei, Y. Tai, and F. Zhao (2012). "The Optimum Design of Power Distribution for Pressurized Water Reactor". In: *Annals of Nuclear Energy* 50, pp. 126–132 (pp. 101, 211).

Darwin, C. R. (1859). *On the Origin of Species*. 1st. London: John Murray (p. 100).

Daubert, C. A., D. J. Kropaczek, and R. E. William (2003). "Fuel Rod Optimization for Coupled BWR Fuel Bundle and Core Design". In: *Proceedings of the Advances of Nuclear Fuel Management III (ANFM-III)*. Hilton Head, South Carolina, USA: American Nuclear Society (ANS) (p. 206).

De Bruyn, D., P. Baeten, S. Larmignat, A. Woaye Hune, and L. Mansani (2010). "The FP7 Central Design Team Project: Towards a Fast-Spectrum Transmutation Experimental Facility". In: *Proceedings of the International Congress on Advances in Nuclear Power Plants 2010 (ICAPP'10)*. 10114. San Diego, California, USA (p. 44).

De Bruyn, D., S. Larmignat, A. Woaye Hune, L. Mansani, G. Rimpault, and C. Artioli (2010). "Accelerator Driven Systems for Transmutation: Main Design Achievements of the XT-ADS and EFIT Systems within the FP6 IP-EUROTRANS Integrated Project". In: *Proceedings of the International Congress on Advances in Nuclear Power Plants 2010 (ICAPP'10)*. 10112. 13–17 June. San Diego, California, USA (p. 44).

De Bruyn, D., H. Aït Abderrahim, P. Baeten, R. Fernandez, J. Engelen, and G. Van den Eynde (2014). "The MYRRHA ADS Project in Belgium enters the Front End Engineering Phase". In: *Proceedings of the International Congress on Advances in Nuclear Power Plants 2014 (ICAPP'14)*. 14012. Charlotte, North Carolina, USA (p. 44).

De Bruyn, D., J. Engelen, A. Orden, and M. P. Aguado (2012). "The Fast-Spectrum Transmutation Experimental Facility FASTEF: Main Design Achievements (Part 2: Reactor Building Design & Plant Layout) within the FP7-CDT Collaborative Project of the European Commission". In: *Proceedings of the International Congress on Advances in Nuclear Power Plants 2012 (ICAPP'12)*. 12015. Chicago, Illinois, USA (p. 44).

De Bruyn, D., R. Fernandez, L. Mansani, M. Sarotto, and E. Bubelis (2012). "The Fast-Spectrum Transmutation Experimental Facility FASTEF: Main Design Achievements (Part 1: Core & Primary Design) within the FP7-CDT Collaborative Project of the European Commission". In: *Proceedings of the International Congress on Advances in Nuclear Power Plants 2012 (ICAPP'12)*. 12014. Chicago, Illinois, USA (p. 44).

De Bruyn, D. and D. Maes (2007). "From MYRRHA to XT-ADS: The Design Evolution of an Experimental ADS Facility". In: *Proceeding of the International Topical Meeting on Nuclear Applications of Accelerators 2007 (AccApp'07)*. Pocatello, Idaho, USA, pp. 848–854 (p. 44).

De Jong, A. J. (1995). *Reloading Pattern Design for Batch Refuelled Nuclear Reactors*. Technical Report IRI-131-95-010. Delft, Netherlands: Delft University of Technology (p. 201).

De Lima, A. M. M., M. D. Machado, J. A. C. C. Medeiros, and R. Schirru (2007). "Study Of Heuristics in Ant System for Nuclear Reload Optimisation". In: *Proceedings of the International Nuclear Atlantic Conference*. Brazil: INAC (pp. 124, 202).

De Lima, A. M. M., R. Schirru, F. C. da Silva, and J. A. C. C. Medeiros (2008). "A Nuclear Reactor Core Fuel Reload Optimization using Artificial Ant Colony Connective Networks". In: *Annals of Nuclear Energy* 35.9, pp. 1606–1612 (pp. 31, 121, 123, 124, 208).

DeChaine, M. D. and M. A. Feltus (1995a). "Comparison of Genetic Algorithm Methods for Fuel Management Optimisation". In: *Proceedings of The International Conference for Mathematics and Computations, Reactor Physics and Environmental Analysis*, pp. 645–651 (pp. 100, 101, 103, 111, 114, 201).

DeChaine, M. D. and M. A. Feltus (1995b). "Nuclear Fuel Management Optimization using Genetic Algorithms". In: *Nuclear Technology* 111.1, pp. 109–114 (pp. 109, 110, 201).

DeChaine, M. D. and M. A. Feltus (1996). "Fuel Management Optimization using Genetic Algorithms and Expert Knowledge". In: *Nuclear Science and Engineering* 124, pp. 188–196 (pp. 103, 107, 109, 111–113, 201).

DeChaine, M. D. (1995). "Stochastic Fuel Management Optimization using Genetic Algorithms and Heuristic Rules". PhD Dissertation. Pennsylvania, USA: Pennsylvania State University (pp. 31, 201).

Derstine, K. L. (1984). *DIF3D: A Code to Solve One-, Two- and Three-Dimensional Finite-Difference Diffusion Theory Problems*. ANL-82-64. Argonne National Laboratory. Argonne, Illinois, USA (p. 66).

Do, B. Q. and L. P. Nguyen (2007). "Application of a Genetic Algorithm to the Fuel Reload Optimization for a Research Reactor". In: *Applied Mathematics and Computation* 187.2, pp. 977–988 (pp. 27, 111, 208).

Do, B. Q., G. Roh, and H. Choi (2006a). "Optimal Refueling Pattern Search for a CANDU Reactor using a Genetic Algorithm". In: *Proceedings of the 2006 International Congress on Advances in Nuclear Power Plants—ICAPP'06*. American Nuclear Society. Reno, Nevada, USA: American Nuclear Society, pp. 2422–2431 (pp. 101, 207).

Do, B. Q., G. Roh, and H. Choi (2006b). "Optimization of a Refueling Simulation for a CANDU Reactor by using an Evolutionary Algorithm". In: *Transactions of the American Nuclear Society* 94, pp. 388–389 (pp. 101, 208).

Dorigo, M., V. Maniezzo, and A. Colorni (1996). "Ant System: Optimization by a Colony of Cooperating Agents". In: *Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 26.1, pp. 29–41 (p. 120).

Dorigo, M., G. Di Caro, and L. M. Gambardella (1999). "Ant Algorithms for Discrete Optimization". In: *Artificial Life* 5.2, pp. 137–172 (p. 120).

Dorigo, M. and T. Stützle (2004). *Ant Colony Optimization*. Cambridge, Massachusett, USA: MIT Press, p. 319 (p. 31).

Dorigo, M. and T. Stützle (2010). "Handbook of Metaheuristics". In: ed. by M. Gendreau and J.-Y. Potvin. Vol. 146. International Series in Operations Research & Management Science. Springer. Chap. Ant Colony Optimization: Overview and Recent Advances, pp. 227–263 (p. 120).

Downar, T. J. and Y. J. Kim (1986). "A Reverse Depletion Method for Pressurized Water Reactor Core Reload Design". In: *Nuclear Technology* 73.1, pp. 42–54 (p. 198).

Downar, T. J. and A. Sesonske (1988). "Light Water Reactor Fuel Cycle Optimization: Theory Versus Practice". In: *Advances in Nuclear Science and Technology* 20, p. 71 (pp. 19, 27, 29, 198).

Driscoll, M. J., T. J. Downar, and E. E. Pilat (1990). *The Linear Reactivity Model for Nuclear Fuel Management*. American Nuclear Society, p. 233 (pp. 19, 21, 213).

Duderstadt, J. J. and L. J. Hamilton (1976). *Nuclear Reactor Analysis*. USA: John Wiley & Sons, Inc. (p. 41).

Erdoğan, A. and M. Geçkinli (2003). "A PWR Reload Optimisation Code (XCore) using Artificial Neural Networks and Genetic Algorithms". In: *Annals of Nuclear Energy* 30.1, pp. 35–53 (pp. 29, 206).

Esquivel-Estrada, J., J. J. Ortiz-Servin, J. A. Castillo, and R. Perusquía (2011). "Azcaxalli: A System Based on Ant Colony Optimization Algorithms, Applied to Fuel Reloads Design in a Boiling Water Reactor". In: *Annals of Nuclear Energy* 38.1, pp. 103–111 (pp. 31, 121, 122, 124, 210).

Fadaei, A. H. and S. Setayeshi (2008). "LONSA as a Tool for Loading Pattern Optimization for VVER-1000 using Synergy of a Neural Network and Simulated Annealing". In: *Annals of Nuclear Energy* 35.10, pp. 1968–1973 (pp. 29, 208).

Fadaei, A. H., N. M. Moghaddam, E. Zahedinejad, M. M. Fadaei, and S. Kia (2010). "Fuel Management Optimization Based on Power Profile by Cellular Automata". In: *Annals of Nuclear Energy* 37.12, pp. 1712–1722 (p. 209).

Fadaei, A. H. and S. Setayeshi (2009). "A New Optimization Method Based on Cellular Automata for VVER-1000 Nuclear Reactor Loading Pattern". In: *Annals of Nuclear Energy* 36.5, pp. 659–667 (p. 209).

Fadaei, A. H., S. Setayeshi, and S. Kia (2009). "An Optimization Method Based on Combination of Cellular Automata and Simulated Annealing for VVER-1000 NPP Loading Pattern". In: *Nuclear Engineering and Design* 239.12, pp. 2800–2808 (pp. 31, 209).

Fagan, J. R. and A. Sesonske (1969). "Optimal Fuel Replacement in Reactivity Limited Systems". In: *Journal of Nuclear Energy* 23.11–12, pp. 683–696 (p. 196).

Fogel, L. J., A. J. Owens, and M. J. Walsh (1966). *Artificial Intelligence through Simulated Evolution*. John Wiley & Sons, Inc. (p. 31).

François, J.-L. and H. A. López (1999). "SOPRAG: A System for Boiling Water Reactors Reload Pattern Optimization using Genetic Algorithms". In: *Annals of Nuclear Energy* 26.12, pp. 1053–1063 (pp. 101, 203).

François, J.-L., C. Martín del Campo, C. C. Cortés, E. Ramírez, and J. Arellano (1999). "Development of an Automated System for Fuel Reload Patterns Design". In: *Nuclear Engineering and Design* 193.193, pp. 13–21 (p. 203).

François, J.-L., J.-R. Guzmáin, C. Martín-del-Campo, and M.-Á. Palomera (2011). "Design and Optimization of an Equilibrium Reload with MOX Fuel with Minor Actinides". In: *Progress in Nuclear Energy* 53.6, pp. 566–570 (p. 210).

François, J.-L., J. J. Ortiz-Servin, C. Martín-del-Campo, A. Castillo, and J. Esquivel-Estrada (2013). "Comparison of Metaheuristic Optimization Techniques for BWR Fuel Reloads Pattern Design". In: *Annals of Nuclear Energy* 51, pp. 189–195 (pp. 101, 212).

Galperin, A. (1995). "Exploration of the Search Space of the In-Core Fuel Management Problem by Knowledge-Based Techniques". In: *Nuclear Science and Engineering* 119.2, pp. 144–152 (pp. 28, 30, 201).

Galperin, A., S. Kimhi, and M. Segev (1989). "A Knowledge-Based System for Optimization of Fuel Reload Configurations". In: *Nuclear Science and Engineering* 102, p. 43 (pp. 30, 199).

Galperin, A. and Y. Kimhy (1991). "Application of Knowledge-Based Methods to In-Core Fuel Management". In: *Nuclear Science and Engineering* 109.1, pp. 103–110 (pp. 30, 199).

Galperin, A. and E. Nissan (1988). "Application of a Heuristic Search Method for Generation of Fuel Reload Configurations". In: *Nuclear Science and Engineering* 99.4, pp. 343–352 (p. 198).

Gen, M. and R. Cheng (1996). "A Survey of Penalty Techniques in Genetic Algorithms". In: *Proceedings of IEEE International Conference on Evolutionary Computation, 1996*. Nagoya, Japan, pp. 804–809 (pp. 153, 154).

GIF (2002). *A Technology Roadmap for Generation IV Nuclear Energy Systems*. Technical Report. Generation IV International Forum (GIF) (p. 3).

GIF (2014). *Technology Roadmap Update for Generation IV Nuclear Energy Systems*. Technical Report. Generation IV International Forum (GIF) (p. 3).

Giraud, B., S. Ehster, G. Locatelli, L. Cinotti, L. Mansani, J. Pirson, X. Jardi, M. T. Dominguez, K. Peers, R. Sunderland, D. Coors, T. Abram, G. Granget, G. Rimpault, E. Gonzalez, A. Mueller, H. Klein, H. Wider, F. Bianchi, D. Struwe, P. Pierini, J. Cetnar, P. Coddington, A. Hogenbirk, B. R. Sehgal, H. Aït Abderrahim, J. Martinez-Val, V. Moreau, P. Vaz, D. Vandeplassche, and P. Phlippen (2005). *Preliminary Design Study on an Experimental Accelerator-Driven System—Final Report*. Technical Report PDS-XADS DOC 05/378. FRAMATOME ANP (p. 44).

Glover, F. W. (1986). "Future Paths for Integer Programming and Links to Artificial Intelligence". In: *Computers and Operations Research* 13.5, pp. 533–549 (p. 30).

Glover, F. W. and M. Laguna (1997). *Tabu Search*. Boston: Springer (p. 31).

Goertzel, G. (1956). "Minimum Critical Mass and Flat Flux". In: *Journal of Nuclear Energy* 2, p. 193 (p. 196).

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. 1st. Addison-Wesley Professional (p. 31).

Goldschmidt, P. (1972). "Minimum Critical Mass in Intermediate Reactors Subject to Constraints on Power Density and Fuel Enrichment". In: *Nuclear Science and Engineering* 49, p. 263 (p. 196).

Gong, Z., K. Wang, and D. Yao (2011). "An Interval Bound Algorithm of Optimizing Reactor Core Loading Pattern by using Reactivity Interval Schema". In: *Annals of Nuclear Energy* 38.12, pp. 2787–2796 (p. 210).

Graves Jr., H. W. (1979). *Nuclear Fuel Management*. ISBN 0-471-03136-4. John Wiley & Sons, Inc., p. 327 (pp. 4, 12, 15, 17, 18).

Griffiths, A. J. F., S. R. Wessler, S. B. Carroll, and J. Doebley (2010). *Introduction ot Genetic Analysis*. 10th. W. H. Freeman (p. 102).

Guler, C., S. Levine, K. Ivanov, J. Svarny, V. Krysl, P. Mikolas, and S. J. (2004). "Development of the VVER Core Loading Optimization System". In: *Annals of Nuclear Energy* 31.7, pp. 747–772 (pp. 101, 207).

Guler, C. (2002). "Development of VVER Core Loading Optimization System". MSc Thesis. Pennsylvania, USA: Pennsylvania State University (p. 205).

Haeck, W. and B. Verboomen (2007). "An Optimum Approach to Monte Carlo Burnup". In: *Nuclear Science and Engineering* 156.2, pp. 180–196 (p. 7).

Haghighattalab, A., A. Minuchehr, A. Zolfaghari, and F. Khoshahval (2012). "Bayesian Inference Along Markov Chain Monte Carlo Approach for PWR Core Loading Pattern Optimization". In: *Annals of Nuclear Energy* 50, pp. 150–157 (p. 211).

Hairer, E. and G. Wanner (1980). *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer-Verlag (p. 7).

Hamasaki, M. and T. Takeda (1986). "Application of Depletion Perturbation Theory to Fuel Loading Optimization". In: *Journal of Nuclear Science and Technology* 23.1, pp. 1–10 (p. 198).

Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation*. 2nd. Prentice Hall, p. 842 (p. 29).

Hays, R. D. (2009). "Boiling Water Reactor In-Core Fuel Management Through Parallel Simulated Annealing in FORMOSA-B". MSc Thesis. Raleigh, North Carolina, USA: North Carolina State University (pp. 184, 209).

Hébert, A. (2009). *Applied Reactor Physics*. Ed. by J. Yelon. Canada: Presses Internationales Polytechnique (pp. 36–38, 41, 42).

Hedayat, A., H. Davilu, A. A. Barfrosh, and K. Sepanloo (2009). "Optimization of the Core Configuration Design using a Hybrid Artificial Intelligence Algorithm for Research Reactors". In: *Nuclear Engineering and Design* 239.12, pp. 2786–2799 (pp. 29, 209).

Henryson II, H., B. J. Toppel, and C. G. Stenberg (1976). *MC^2-2: A Code to Calculate Fast Neutron Spectra and Multigroup Cross Sections*. ANL-8144. Technical Report. ANL-8144. Argonne National Laboratory (ANL). Argonne, Illinois, USA (p. 64).

Hill, N. J. and G. T. Parks (2015). "Pressurized Water Reactor In-Core Nuclear Fuel Management by Tabu Search". In: *Annals of Nuclear Energy* 75, pp. 64–71 (pp. 31, 101, 213).

Ho, A. L. B. (1981). "Extended Burnup Fuel Cycle Optimization for Pressurized Water Reactors". PhD Dissertation. Lafayette, Indiana, USA: Purdue University (p. 197).

Ho, A. L. B. and A. Sensoke (1982). "Extended Burnup Fuel Cycle Optimization for Pressurized Water Reactors". In: *Nuclear Technology* 58, p. 422 (p. 197).

Ho, L.-W. and A. F. Rohach (1982). "Perturbation Theory in Nuclear Fuel Management Optimization". In: *Nuclear Science and Engineering* 82.2, pp. 151–161 (p. 197).

Hoareau, F. (2008). "Loading Pattern Optimization using Ant Colony Algorithm". In: *Proceedings of the International Conference on the Physics of Reactors Nuclear Power (PHYSOR'08): A Sustainable Resource*. Casino-Kursaal Conference Center, Interlaken, Switzerland (pp. 31, 121, 122, 124, 208).

Hobson, G. H. and P. J. Turinsky (1986). "Automatic Determination of Pressurized Water Reactor Core Loading Patterns That Maximize Beginning-of-Cycle Reactivity Within Power-Peaking and Burnup Constraints". In: *Nuclear Technology* 74.1, pp. 5–13 (p. 198).

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. 1st. Ann Arbor, Michigan, USA: University of Michigan Press (pp. 31, 100, 103, 104, 115).

Hongchun, W. (2001). "Pressurized Water Reactor Reloading Optimization using Genetic Algorithms". In: *Annals of Nuclear Energy* 28.13, pp. 1329–1341 (pp. 103, 205).

Hoshino, T. (1972). "In-Core Fuel Management Optimization by Heuristic Learning Technique". In: *Nuclear Scince and Engineering* 49, pp. 59–67 (p. 196).

Hosseini, M. and N. Vosoughi (2012). "Development of a VVER-1000 Core Loading Pattern Optimization Program Based on Perturbation Theory". In: *Annals of Nuclear Energy* 39.1, pp. 35–41 (p. 211).

Hosteny, R. P. (1978). *The ARC System Fuel Cycle Analyses Capability, REBUS-2*. ANL-7721. Argonne National Laboratory (ANL). Argonne, Illinois, USA (pp. 67, 81).

Howland, H. R., C. N. Dorny, and R. L. Stover (1973). "Power Flattening and Loading Pattern Searches for Pressurized Water Reactors". In: *Transactions of American Nuclear Society*, pp. 170–171 (p. 196).

Huang, H.-Y. (1979). "Optimal Fuel Management by Maximizing the End-of-Cycle Effective Multiplication Factor". PhD Dissertation. Philadelphia, Pennsylvania, USA: Pennsylvania State University (p. 197).

Huang, H.-Y. and S. Levine (1978). "A New Method Optimizing Core Reloads". In: *Transactions of the American Nuclear Society* 30, p. 339 (p. 197).

IAEA (1995). *In-Core Fuel Management: Reloading Techniques—Proceedings of a Technical Committee meeting and Workshop held in Vienna, 19–21 October 1992*. Technical Report IAEA-TECDOC-816. Vienna, Austria: International Atomic Energy Agency (IAEA) (p. 21).

IAEA (1996). *Defence in Depth in Nuclear Safety: INSAG-10*. Technical Report STI/PUB/1013. Vienna, Austria: International Atomic Energy Agency (IAEA) (p. 13).

IAEA (2007). *Estimation of Global Inventories of Radioactive Waste and Other Radioactive Materials*. Technical Report IAEA-TECDOC-1591. Vienna, Austria: International Atomic Energy Agency (IAEA) (p. 2).

IAEA (2009). *Advanced Reactor Technology Options for Utilization and Transmutation of Actinides in Spent Nuclear Fuel*. Technical Report IAEA-TECDOC-1626. Vienna, Austria: International Atomic Energy Agency (IAEA) (pp. 4, 5).

IAEA (2013a). *Energy, Electricity and Nuclear Power Estimates for the Period up to 2050*. Vienna, Austria (p. 2).

IAEA (2013b). *Nuclear Power Reactors in the World*. Vienna, Austria (p. 2).

Iqbal, M., S. M. Mirza, and M. S. Ali (2008). "MTR Core Loading Pattern Optimization using Burnup Dependent Group Constants". In: *Nuclear Technology & Radiation Protection* 23.2, pp. 28–33 (pp. 27, 208).

Ishida, S. and H. Sekimoto (2010a). "Applicability of Dynamic Programming to the Accelerator-Driven System (ADS) Fuel Cycle Shuffling Scheme for Minor Actinide (MA) Transmutation". In: *Annals of Nuclear Energy* 37.3, pp. 406–411 (pp. 27, 210).

Ishida, S. and H. Sekimoto (2010b). "Finding the Best Fuel Assemblies Shuffling Scheme of ADS for MA Transmutation using Dynamic Programming". In: *Nuclear Engineering and Design* 240.10, pp. 3645–3653 (pp. 27, 210).

Isotalo, A. E. and P. A. Aarnio (2011). "Comparison of Depletion Algorithms for Large Systems of Nuclides". In: *Annals of Nuclear Energy* 38.2–3, pp. 261–268 (p. 41).

Izenson, M. G. (1983). "Automated PWR Reload Design Optimization". MSc Thesis. Massachusetts, USA: Massachusetts Institute of Technology, Department of Nuclear Engineering (p. 197).

Jagawa, S., T. Yoshii, and A. Fukao (2001). "Boiling Water Reactor Loading Pattern Optimization using Simple Linear Perturbation and Modified Tabu Search Methods". In: *Nuclear Science and Engineering* 138.1, pp. 67–77 (pp. 31, 205).

Jaluvka, D., G. Van den Eynde, and S. Vandewalle (2012). "Development of a Core Management Tool for MYRRHA: Work in Progress". In: *Proceedings of the 2012 20th International Conference on Nuclear Engineering collocated with the ASME 2012 Power Conference ICONE20-POWER2012*. Vol. 2. Anaheim, California, USA: American Society of Mechanical Engineers (ASME), pp. 625–631 (pp. i, 211).

Jaluvka, D., G. Van den Eynde, and S. Vandewalle (2013a). "Development of a Core Management Tool for MYRRHA". In: *Energy Conversion and Management* 74. ISSN 0196-8904, pp. 562–568 (p. i).

Jaluvka, D., G. Van den Eynde, and S. Vandewalle (2013b). "MYRRHA Core Loading Pattern Optimization using Metaheuristics". In: *Book of Abstracts of the 16th International Conference on Emerging Nuclear Energy Systems (ICENES)*. Madrid, Spain (pp. i, 212).

Jamalipour, M., M. Gharib, R. Sayareh, and F. Khoshahval (2013). "PWR Power Distribution Flattening using Quantum Particle Swarm Intelligence". In: *Annals of Nuclear Energy* 56, pp. 143–150 (p. 212).

Jamalipour, M., R. Sayareh, M. Gharib, F. Khoshahval, and M. R. Karimi (2013). "Quantum Behaved Particle Swarm Optimization with Differential Mutation Operator Applied to WWER-1000 In-Core Fuel Management Optimization". In: *Annals of Nuclear Energy* 54, pp. 134–140 (pp. 31, 212).

Jang, C. S., H. J. Shim, and C. H. Kim (2001). "Optimization Layer by Layer Networks for In-Core Fuel Management Optimization Computations in PWRs". In: *Annals of Nuclear Energy* 28.11, pp. 1115–1132 (p. 205).

Jiang, S., A. K. Ziver, J. N. Carter, C. C. Pain, A. J. H. Goddard, S. Franklin, and H. J. Phillips (2006). "Estimation of Distribution Algorithms for Nuclear Reactor Fuel Management Optimisation". In: *Annals of Nuclear Energy* 33.11-12, pp. 1039–1057 (p. 208).

Jiang, S. (2009). "Nuclear Fuel Management Optimisation using Estimation of Distribution Algorithms". PhD Dissertation. London, UK: Imperial College London, University of London (p. 209).

Johansen, B. J. (1994). "ALPS: An Advanced Interactive Fuel Management Package". In: *Proceeding of the Topical Meeting on Advances in Reactor Physics*. Vol. 3. Knoxville, Tennessee, USA, p. 324 (p. 174).

Jongen, Y. (1999). "ADONIS: The Proton-Driven Neutron Source for Radioisotope Production". In: *Production Technologies for Molybdenum-99 and Technetium-99m*. IAEA-TECDOC-1065. Vienna, Austria: International Atomic Energy Agency (IAEA), pp. 139–146 (p. 44).

Kalcheva, S. and E. Koonen (2012). "MYRRHA, a Flexible Fast Spectrum Irradiation Facility". In: *Transactions of the 34th International Meeting on Reduced Enrichment for Research and Test Reactors (RERTR 2012)*. Warsaw, Poland (pp. 15, 26, 211).

Karahroudi, M. R., S. A. M. Shirazi, and K. Sepanloo (2013). "Optimization of Designing the Core Fuel Loading Pattern in a VVER-1000 Nuclear Power Reactor using the Genetic Algorithm". In: *Annals of Nuclear Energy* 57, pp. 142–150 (pp. 101, 212).

Karve, A. A., P. M. Keller, P. J. Turinsky, and G. I. Maldonado (2001). "Nuclear Fuel Management Optimization". In: *Transactions of the American Nuclear Society* 84 (p. 205).

Karve, A. A. and P. J. Turinsky (2000). "FORMOSA-B: A Boiling Water Reactor In-Core Fuel Management Optimization Package II". In: *Nuclear technology* 131.1, pp. 48–68 (p. 204).

Karve, A. A. and P. J. Turinsky (2001). "FORMOSA-B: A Boiling Water Reactor In-Core Fuel Management Optimization Package III". In: *Nuclear technology* 135.3, pp. 241–251 (p. 205).

Kashi, S., A. Minuchehr, N. Poursalehi, and A. Zolfaghari (2014). "Bat Algorithm for the Fuel Arrangement Optimization of Reactor Core". In: *Annals of Nuclear Energy* 64, pp. 144–151 (p. 213).

Kato, Y., M. Odamura, H. Urushihara, and H. Matsushima (1999). "Decomposition Principle for Refueling Optimization in Fast Breeder Reactors". In: *Nuclear Science and Engineering* 133.2, pp. 119–146 (pp. 27, 203).

Keller, P. M. and P. J. Turinsky (1997). "Recent Developments in FORMOSA-P". In: *Advances in Nuclear Fuel Management II*. Myrtle Beach: American Nuclear Society (ANS), pp. 8–39 (p. 202).

Kennedy, J. and R. C. Eberhart (1995). "Particle Swarm Ooptimization". In: *IEEE International Conference on Neural Networks (ICNN'95)*. Vol. 4. IEEE. Perth, Western Australia (p. 31).

Keyvani, M., M. Arkani, and A. Hossni Rokh (2010). "Optimization of In-Core Fuel Management Strategy of Tehran Research Reactor (TRR) using MCNP-4C". In: *Annals of Nuclear Energy* 37.12, pp. 1683–1687 (pp. 27, 210).

Khoshahval, F. (2009). "Fuel Managment Optimization of Bushehr WWER-1000 Reactor Core". MSc Thesis. Iran: Shahid Beheshti University (p. 209).

Khoshahval, F., H. Minuchehr, and A. Zolfaghari (2011). "Performance Evaluation of PSO and GA in PWR Core Loading Pattern Optimization". In: *Nuclear Engineering and Design* 241.3, pp. 799–808 (pp. 31, 101, 210).

Khoshahval, F., A. Zolfaghari, H. Minuchehr, M. Sadighi, and A. Norouzi (2010). "PWR Fuel Management Optimization using Continuous Particle Swarm Intelligence". In: *Annals of Nuclear Energy* 37.10, pp. 1263–1271 (p. 210).

Kim, H. C., M. Y. Hsiao, and S. H. Levine (1989). "Increased-Discharge-Burnup Method for Multicycle Reload Design". In: *Nuclear Technology* 86.3, pp. 289–304 (p. 199).

Kim, H. G., S. H. Chang, and B. H. Lee (1993). "Optimal Fuel Loading Pattern Design using an Artificial Neural Network and a Fuzzy Rule-Based System". In: *Nuclear Science and Engineering* 115.2, pp. 152–163 (p. 200).

Kim, H. (1989). "A Multicycle Reload Optimization Method in Once-Through Fuel Cycles of Light-Water Reactors". PhD Dissertation. University Park, Pennsylvania, USA: Pennsylvania State University (p. 199).

Kim, T. K. and C. H. Kim (1996). "Determination of Optimized PWR Fuel Loading Pattern by Mixed Integer Programming". In: *PHYSOR'96*. Mito, Japan, pp. 176–185 (pp. 28, 202).

Kim, T. K. and C. H. Kim (1997). "Mixed Integer Programming for Pressurized Water Reactor Fuel-Loading-Pattern Optimization". In: *Nuclear Science and Engineering* 127.3, pp. 346–357 (pp. 28, 202).

Kim, Y. J., T. J. Downar, and A. Sesonske (1987). "Optimization of Core Reload Design for Low-Leakage Fuel Management in Pressurized Water Reactors". In: *Nuclear Science and Engineering* 96.2, pp. 85–101 (pp. 30, 198).

Kim, Y. J. (1986). "Optimization of Core Reload Design for Low-Leakage Fuel Management in Pressurized Water Reactors". PhD Dissertation. Indiana, USA: Purdue University (p. 198).

Kinsey, R. (1983). *ENDF-102—Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF/B-V*. Technical Report. 2nd revised edition. New York, USA: Brookhaven National Laboratory (BNL), pp. 2887–2996 (pp. 64, 84).

Kirkpatrick, S., C. D. Gellat, and M. P. Vecchi (1983). "Optimization by Simulated Annealing". In: *Science* 220, pp. 671–680 (p. 31).

Klerk, E. de, C. Roos, T. Terlaky, T. Illés, A. J. de Jong, J. Valkó, and J. E. Hoogenboom (1997). "Optimization of Nuclear Reactor Reloading Patterns". In: *Annals of Operations Research* 69, pp. 65–84 (pp. 28, 202).

Knott, D. and C. Oyarzun (2011). "Description of the Shuffle Design Software MARLA". In: *Progress in Nuclear Energy* 53.6, pp. 607–617 (p. 210).

Kobayashi, Y., S. Kondo, and Y. Togo (1976). "Optimization of Refueling Scheme of Sodium-Cooled Fast Reactor Core using Heuristic Enumeration Method". In: *Journal of Nuclear Science and Technology* 13.9, pp. 471–482 (pp. 27, 197).

Kobayashi, Y., S. Kondo, and Y. Togo (1977). "Optimization of Refueling Scheme of Fast Reactor Core by Means of Man Machine Interactive Method". In: *Journal of Nuclear Science and Technology* 14.3, pp. 177–194 (pp. 27, 197).

Kobayashi, Y. and E. Aiyoshi (2002). "Optimization of Boiling Water Reactor Loading Pattern using Two-Stage Genetic Algorithm". In: *Nuclear Science and Engineering* 142.2, pp. 119–139 (pp. 101, 205).

Kobayashi, Y. and E. Aiyoshi (2003a). "Inclusion of Interactive GA in the Automatic Core Design of a BWR". In: *Advances in Nuclear Fuel Management III (ANFM III)*. Hilton Head Island, South Carolina, USA: American Nuclear Society (ANS) (p. 206).

Kobayashi, Y. and E. Aiyoshi (2003b). "Optimization of Boiling Water Reactor Loading Pattern using an Improved Genetic Algorithm". In: *Nuclear Technology* 143.2, pp. 144–151 (pp. 31, 101, 206).

Koolwaaij, J. W. (1995). *Optimization of the Reload Pattern for a Nuclear Reactor using Perturbation Theory*. Report. Delft, Nederland: Technische Universiteit Delft (p. 201).

Koza, J. R. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, Massachusetts, USA: MIT Press (p. 31).

Kropaczek, D. J. (1989). "In-Core Nuclear Fuel Management Optimization Utilizing Simulated Annealing". PhD Dissertation. North Carolina, USA: North Carolina State University (pp. 31, 199).

Kropaczek, D. J., J. McElroy, and P. J. Turinsky (1993). "Validity of Single-Cycle Objective Functions for Multicycle Reload Design Optimization". In: *Transactions of the American Nuclear Society* 69, pp. 419–420 (p. 200).

Kropaczek, D. J., G. T. Parks, G. I. Maldonado, and P. J. Turinsky (1991). "Application of Simulated Annealing to In-Core Nuclear Fuel Management Optimization". In: *Proceedings of the International Top Meeting on Advances in Mathematics, Computations and Reactor Physics*. Pittsburgh, Pennsylvania, USA: American Nuclear Society (p. 199).

Kropaczek, D. J. and P. J. Turinsky (1990). "A Higher Order Depletion Perturbation Theory with Application to In-Core Fuel Management Optimization". In: *Transactions of the American Nuclear Society* 61. Conference: American Nuclear Society (ANS) annual meeting, pp. 362–363 (p. 199).

Kropaczek, D. J. and P. J. Turinsky (1991). "In-Core Nuclear Fuel Management Optimization for Pressurized Water Reactors Utilizing Simulated Annealing". In: *Nuclear Technology* 95.1, pp. 9–32 (pp. 31, 199).

Kropaczek, D. J., P. J. Turinsky, G. T. Parks, and G. I. Maldonado (1994). "The Efficiency and Fidelity of the In-Core Nuclear Fuel Management Code FORMOSA-P". In: *Proceedings of the International Conference on Reactor Physics and Reactor Computations*. Tel Aviv, Israel: Ben-Gurion University of the Negev Press, p. 572 (p. 200).

Krüger, P. P. (2004). "Numerical Methods to Solve the Fuel Depletion Equations for a Nuclear Reactor". MSc Thesis. Potchefstroom, South Africa: Potchefstroomse Universiteit vir Christelike Ho er Onderwys (p. 41).

Kubokawa, T. and R. Kiyose (1975). "Optimization of In-Core Fuel Management by Integer Linear Programming". In: *Proceedings of the Conference on Computational Methods in Nuclear Engineering*. Vol. 1. Charleston, South Carolina, USA, pp. 135–149 (pp. 28, 197).

Kyung-Eung, K. (1984). "The In-Core Fuel Management by Variational Method". In: *Journal of the Korean Nuclear Society* 16.4, pp. 181–194 (p. 198).

Lam, H. Q. and et al. (2003). "LP-Fun Applications to Reload Design". In: *Proceedings on the Advances in Nuclear Fuel Management III (ANFM-III)*. Hilton Head, South Caroline, USA: American Nuclear Society (p. 206).

Lamarsh, J. R. (1972). *Introduction to Nuclear Reactor Theory*. Ed. by H. Goldstein. Second printing. Nuclear Engineering. Addison-Wesley Publishing Company, Inc. (p. 39).

Lawrence, R. D. (1983). *The DIF3D Nodal Neutronics Option for Two- and Three-Dimensional Diffusion Theory Calculations in Hexagonal Geometry*. ANL-83-1. Argonne National Laboratory. Argonne, Illinois, USA (p. 66).

Lee, C. H. and W. S. Yang (2007). *Status Reprot on Multigroup Cross Section Generation Code Development for High-Fidelity Deterministic Neutronics Simulation System*. Technical Report ANL-AFCI-207. Argonne, Illinois, USA: Argonne National Laboratory (ANL) (p. 64).

Lee, H. C., H. J. Shim, and C. H. Kim (1999). "Hybrid Neutronics Evaluation Model Incorporated into Parallel Computing Simulated Annealing Fuel Loading Pattern Optimization Schemes". In: *International Conference on Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*. Madrid, Spain (p. 204).

Lee, H. C., H. J. Shim, and C. H. Kim (2001). "Parallel Computing Adaptive Simulated Annealing Scheme for Fuel Assembly Loading Pattern Optimization in PWRs". In: *Nuclear Technology* 135.1, pp. 39–50 (pp. 31, 184, 205).

Levine, S. H. (1986). "Handbook of Nuclear Reactor Calculations". In: ed. by Y. Ronen. Vol. II. Boca Raton, Florida, USA: CRC Press, Inc. Chap. In-Core Fuel Management of Four Reactor Types (pp. 13, 17, 34, 198).

Levine, S. H. (1991). "Optimization Techniques Used to Determine Superior Reload Configurations for the TMI-1 Core". In: *Proceedings of the ANS International Topical Meeting on Advances in Mathematics, Computations, and Reactor Physics*. Pittsburgh, Pennsylvania, USA (p. 199).

Levine, S. H. (1995). "History and Status of Reloading Techniques for Light Water Reactors". In: *In-Core Fuel Management: Reloading Techniques—Proceedings of a Technical Committee meeting and Workshop held in Vienna, 19–21 October 1992*. IAEA-TECDOC-816. International Atomic Energy Agency (IAEA). Vienna, Austria: International Atomic Energy Agency (IAEA), pp. 15–61 (pp. 27, 201).

Levine, S. H. and D. Bai (1993). "PFMP-MCSO, a Code That Performs Automatically Multicycle Reloads with Optimum Single-Cycle Patterns". In: *Transactions of the American Nuclear Society* 69, pp. 417–418 (p. 200).

Levine, S. H., D. Bai, and L. Zhian (1994). "The Ultimate In-Core Fuel Management Code for Pressurized Water Reactors". In: *Proceedings of the International Conference on Reactor Physics and Reactor Computations*. Ed. by Y. Ronen and E. Elias. Vol. 25. 14. Tel Aviv, Israel, pp. 552–558 (p. 200).

Levine, S. H., J. S. John, and L. Zhian (1994). "Practical Implications of using Automatic Optimization Codes for Reloading Nuclear Reactors". In: *Proceedings of the Topical Meeting on Advances in Reactor Phyiscs*. Vol. 1. American Nuclear Society. Knoxville, Tennessee, USA, p. 308 (p. 200).

Lewis, E. E. and W. F. Miller Jr. (1993). *Computational Methods of Neutron Transport*. La Grange Park, Illinois, USA: American Nuclear Society (ANS) (p. 66).

Leyffer, S. (2013). *Private Communication* (pp. 25, 28).

Li, Z. and S. H. Levine (1994). "AUTOLOAD, An Automatic Optimal Pressurized Water Reactor Reload Design System with an Expert Module". In: *Nuclear Science and Engineering* 118.2, pp. 67–78 (p. 200).

Li, Z. (1993). "An Automatic Optimal Pressurized Water Reactor Reload Design Expert System". PhD Dissertation. Philadelphia, Pennsylvania, USA: Pennsylvania State University (pp. 30, 200).

Lim, J. (2011). *Thermodynamic Assessment of the Oxygen Control Boundary for the Operation of MYRRHA*. Technical Report SCK-CEN-I-271. Confidential. Mol, Belgium: SCK • CEN (p. 51).

Lin, B. I., B. Zolotar, and J. Weisman (1979). "Automated Procedure for Selection of Optimal Refueling Policies for Light Water Reactors". In: *Nuclear Technology* 44.2, pp. 258–275 (p. 197).

Lin, C. and B.-F. Lin (2012). "Automatic Pressurized Water Reactor Loading Pattern Design using Ant Colony Algorithms". In: *Annals of Nuclear Energy* 43, pp. 91–98 (pp. 31, 121, 211).

Lin, C., J.-I. Yang, K.-J. Lin, and Z.-D. Wang (1998). "Pressurized Water Reactor Loading Pattern Design using the Simple Tabu Search". In: *Nuclear Science and Engineering* 129.1, pp. 61–71 (pp. 32, 203).

Lin, K.-J. and C. Lin (1998). "Pressurized Water Reactor Reload Design by an Expert System". In: *Nuclear Science and Engineering* 130.1, pp. 128–140 (pp. 30, 203).

Liu, S. and J. Cai (2012). "Studies of Fuel Loading Pattern Optimization for a Typical Pressurized Water Reactor (PWR) using Improved Pivot Particle Swarm Method". In: *Annals of Nuclear Energy* 50, pp. 117–125 (p. 211).

Machado, L. and R. Schirru (2002). "The Ant-Q Algorithm Applied to the Nuclear Reload Problem". In: *Annals of Nuclear Energy* 29.12, pp. 1455–1470 (pp. 31, 121, 123, 124, 206).

Machado, M. D. (1999). "Um Novo Algoritmo Evolucionário com Aprendizagem LVQ para Otimização de Problemas Combinatórios como a Recarga de Reatores Nucleares". MSc Thesis. Brazil: COPPE/UFRJ (p. 204).

Machado, M. D. (2005). "Algoritmo Evolucionário PBIL Multi-Objetivo Aplicado ao Problema da Recarga de Reatores Nucleares". PhD Dissertation. Brazil: COPPE/UFRJ (p. 207).

Maes, D. (2005). "Design of the Small-Scale Experimental ADS: MYRRHA". In: *Proceedings of the 12th International Conference on Emerging Nuclear Energy Systems (ICENES'2005)*. Brussels, Belgium (p. 44).

Mahlers, Y. P. (1994). "Core Loading Pattern Optimization for Pressurized Water Reactors". In: *Annals of Nuclear Energy* 21.4, pp. 223–227 (p. 200).

Mahlers, Y. P. (1995). "Core Loading Pattern Optimization Based on Simulated Annealing and Successive Linear Programming". In: *Annals of Nuclear Energy* 22.1, pp. 29–37 (pp. 31, 201).

Mahlers, Y. P. (1997). "Core Loading Pattern Optimization for Research Reactors". In: *Annals of Nuclear Energy* 27.7, pp. 509–514 (pp. 27, 28, 202).

Mahlers, Y. P. (2002). "Core Reload Optimization for Equilibrium Cycles using Simulated Annealing and Successive Linear Programming". In: *Annals of Nuclear Energy* 29.11, pp. 1327–1344 (p. 206).

Malambu, E. (2014). *Private Communication* (p. 72).

Malambu, E. and R. Popescu (2014). *Private Communication* (p. 83).

Maldonado, G. I. (2005). "Optimizing LWR Cost of Margin One Fuel Pin at a Time". In: *IEEE Transactions on Nuclear Science* 52, pp. 996–1003 (p. 27).

Maldonado, G. I., P. J. Turinsky, D. J. Kropaczek, and G. T. Parks (1995). "Employing Nodal Generalized Perturbation Theory for the Minimization of Feed Enrichment During Pressurized Water Reactor In-Core Nuclear Fuel Management Optimization". In: *Nuclear Science and Engineering* 121.2, pp. 312–325 (p. 201).

Maldonado, G. I. and P. J. Turinsky (1995). "Application of Nonlinear Nodal Diffusion Generalized Perturbation Theory to Nuclear Fuel Reload Optimization". In: *Nuclear Technology* 110.2, pp. 198–219 (p. 201).

Martín del Campo, C., J. L. François, L. Avendaño, and M. González (2004). "Development of a BWR Loading Pattern Design System Based on Modified Genetic Algorithms and Knowledge". In: *Annals of Nuclear Energy* 31.16, pp. 1901–1911 (pp. 101, 207).

Martín del Campo, C., M.-Á. Palomera-Pérez, and J.-L. François (2009). "Advanced and Flexible Genetic Algorithms for BWR Fuel Loading Pattern Optimization". In: *Annals of Nuclear Energy* 36.10, pp. 1553–1559 (p. 209).

Mazrou, H. and M. Hamadouche (2006). "Development of a Supporting Tool for Optimal Fuel Management in Research Reactors using Artificial Neural Networks". In: *Nuclear Engineering and Design* 236.3, pp. 255–266 (p. 208).

Mélice, M. (1969). "Pressurized Water Reactor Optimal Core Management and Reactivity Profile". In: *Nuclear Science and Engineering* 37, pp. 451–477 (p. 196).

Meneses, A. A. d. M. (2005). "Otimização por Enxame de Partículas Aplicado ao Problema Combinatório da Recarga de um Reator Nuclear". MSc Thesis. Brazil: COPPE/UFRJ (p. 207).

Meneses, A. A. d. M. and R. Schirru (2006). "Particle Swarm Optimization Applied to the Combinatorial Problem in Order to Solve the Nuclear Reactor Fuel Reloading Problem". In: *Proceedings of the Seventh International Conference on Applied Artificial Intelligence*. FLINS. Italy (pp. 31, 208).

Meneses, A. A. d. M., L. M. Gambardella, and R. Schirru (2010). "A New Approach for Heuristics-Guided Search in the In-Core Fuel Management Optimization". In: *Progress in Nuclear Energy* 52.4, pp. 339–351 (p. 210).

Meneses, A. A. d. M., A. M. M. de Lima, and R. Schirru (2010). "Nuclear Power". In: ed. by P. Tsvetkov. Sciyo. Chap. Artificial Intelligence Methods Applied to the In-Core Nuclear Fuel Management Optimization, pp. 63–78 (pp. 101, 210).

Meneses, A. A. d. M., M. D. Machado, and R. Schirru (2009). "Particle Swarm Optimization Applied to the Nuclear Reload Problem of a Pressurized Water Reactor". In: *Progress in Nuclear Energy* 51.2, pp. 319–326 (pp. 104, 201).

Meneses, A. A. d. M., P. Rancoita, R. Schirru, and L. M. Gambardella (2010). "A Class-Based Search for the In-Core Fuel Management Optimization of a Pressurized Water Reactor". In: *Annals of Nuclear Energy* 37.11, pp. 1554–1560 (p. 210).

Mingle, J. O. (1973). "In-core Fuel Management using Perturbation Theory and Linear Programming". In: *Transactions of the American Nuclear Society* 17, pp. 306–307 (p. 196).

Mingle, J. O. (1975). "In-Core Fuel Management via Perturbation Theory". In: *Nuclear Technology* 27, pp. 248–257 (p. 197).

Mingle, J. O. (1978). "Nuclear Fuel Management via Fuel Quality Factor Averaging". In: *Annals of Nuclear Energy* 5.11-12, pp. 645–648 (p. 197).

Mishra, S., R. S. Modak, and S. Ganesan (2009). "Optimization of Depleted Uranium Bundle Loading in Fresh Core of Indian PHWR by Evolutionary Algorithm". In: *Annals of Nuclear Energy* 37.2, pp. 208–217 (p. 209).

Moore, B. R., P. J. Turinsky, and A. A. Karve (1999). "FORMOSA-B: A Boiling Water Reactor In-Core Fuel Management Optimization Package". In: *Nuclear technology* 126.2, pp. 153–169 (p. 204).

Morita, T., Y. A. Chao, A. J. Federowicz, and P. J. Duffy (1986). "LPOP: Loading Pattern Optimization Program". In: *Transactions of the American Nuclear Society* 52, pp. 41–42 (p. 198).

Motoda, H., J. Herczeg, and A. Sesonske (1975). "Optimization of Refueling Schedule for Light Water Reactors". In: *Nuclear Technology* 25.3, pp. 477–496 (p. 197).

Motoda, H. and O. Yokomizo (1977). "Method to Minimize Power Peaking in Refueling Schedule of Boiling Water Reactor". In: *Journal of Nuclear Science and Technology* 14, p. 108 (p. 197).

Motoda, H. and O. Yokomizo (1976). "Optimization of Fuel Assembly Allocation for Boiling Water Reactors". In: *Journal of Nuclear Science and Technology* 13.5, pp. 230–246 (p. 197).

Mylonakis, A. (2012). *Private Communication* (p. 70).

SCK•CEN (2014). *MYRRHA: Multi-purpose hYbrid Research Reactor for High-tech Applications*. SCK•CEN. URL: http://myrrha.sckcen.be/ (p. 44).

Naft, B. N. and A. Sesonske (1972). "Pressurized Water Reactor Optimal Fuel Management". In: *Nuclear Technology* 14.2, pp. 123–132 (p. 196).

Nazari, T., M. Aghaie, A. Zolfaghari, A. Minuchehr, and A. Norouzi (2013). "WWER Core Pattern Enhancement using Adaptive Improved Harmony Search". In: *Nuclear Engineering and Design* 254, pp. 23–32 (pp. 32, 212).

Nicolau, A. d. S., R. Schirru, and A. M. M. de Lima (2012). "Nuclear Reactor Reload using Quantum Inspired Algorithm". In: *Progress in Nuclear Energy* 55, pp. 40–48 (p. 211).

Nissan, E. and A. Galperin (1998). "Refueling in Nuclear Engineering: The FUELCON Project". In: *Computers in Industry* 37.1, pp. 43–54 (p. 203).

Norouzi, A., M. Aghaie, A. R. Mohamadi Fard, A. Zolfaghari, and A. Minuchehr (2013). "Nuclear Reactor Core Optimization with Parallel Integer Coded Genetic Algorithm". In: *Annals of Nuclear Energy* 60, pp. 308–315 (pp. 101, 184, 212).

NRG (2014). *In-core Fuel Management Services*. NRG. URL: ww.nrg.eu/safety-power/in-core-fuel-management-services/ (p. 174).

Object Management Group (2014). *Unified Modeling Language (UML) Resource Page*. Object Management Group. URL: http://www.uml.org/ (p. 177).

OECD-NEA (2002). *Accelerator-Driven Systems (ADS) and Fast Reactors (FR) in Advanced Nuclear Fuel Cycles*. Comparative Study. Paris, France: Nuclear Energy Agency (OECD-NEA) (pp. 4, 5).

OECD-NEA (2006). *Advanced Nuclear Fuel Cycles and Radioactive Waste Management*. NEA No. 5990. Paris, France: Nuclear Energy Agency (OECD-NEA) (p. 4).

OECD-NEA (2012). *Uranium 2011: Resources, Production and Demand*. NEA No. 7059. Paris, France: Nuclear Energy Agency (OECD-NEA) (p. 2).

OECD-NEA (2014). *Computer Program Services*. Nuclear Energy Agency (OECD-NEA). URL: http://www.oecd-nea.org/dbprog/ (p. 62).

Ohkawachi, Y., S. Maeda, H. Nagasaki, T. Sekine, and T. Aoyama (2003). *JOYO MK-II Core Characteristics Database*. Research Document JNC TW9409 2003-001. Muramatsu, Japan: Japan Nuclear Cycle Development Institute (p. 85).

Okafor, K. C. and T. Aldemir (1988). "Construction of Linear Empirical Core Models for Pressurized Water Reactor In-Core Fuel Management". In: *Nuclear Technology* 81.3, pp. 381–392 (pp. 28, 198).

Oliveira, I. M. S. de and R. Schirru (2011). "Swarm Intelligence of Artificial Bees Applied to In-Core Fuel Management Optimization". In: *Annals of Nuclear Energy* 38.5, pp. 1039–1045 (pp. 31, 211).

Olson, A. P. (2001). *A Users Guide for the REBUS-PC Code, Version 1.4*. Argonne National Laboratory (ANL). Argonne, Illinois, USA (p. 67).

ORNL (2011). *SCALE: A Comprehensive Modeling and Simulation Suite for Nuclear Safety Analysis and Design*. Version 6.1. ORNL/TM-2005/39. Oak Ridge National Laboratory (ORNL). Oak Ridge, Tennessee, USA (p. 70).

ORNL-RSICC (2014). *Radiation Safety Information Computational Center*. Oak Ridge National Laboratory (ORNL). URL: https://rsicc.ornl.gov/ (p. 64).

Ortiz, J. J., A. Castillo, J. L. Montes, and R. Perusquía (2007). *Comparación de Técnicas de Optimizacin Aplicadas al Diseño de la Recarga de Combustible Nuclear* (p. 208).

Ortiz, J. J. and I. Requena (2004a). "An Order Coding Genetic Algorithm to Optimize Fuel Reloads in a Nuclear Boiling Water Reactor". In: *Nuclear Science and Engineering* 146.1, pp. 88–98 (pp. 101, 207).

Ortiz, J. J. and I. Requena (2004b). "Using a Multi-State Recurrent Neural Network to optimize Loading Patterns in BWRs". In: *Annals of Nuclear Energy* 31.7, pp. 789–803 (p. 207).

Ortiz-Servin, J. J., J. A. Castillo, and D. A. Pelta (2011). "BWR Fuel Cycle Optimization using Neural Networks". In: *Nuclear Engineering and Design* 241.9, pp. 3729–3735 (pp. 29, 211).

Oyarzun, C. (2003). "The Global Nuclear Fuel Optimization System for BWR Fuel Cycle Management". In: *Proceedigns of the Advances in Nuclear Fuel Management III (ANFM-III)*. Hilton Head, South Carolina, USA: American Nuclear Society (p. 206).

Palmiotti, G., E. E. Lewis, and C. B. Carrico (1995). *VARIANT: VARIational Anisotropic Nodal Transport for Multidimensional Cartesian and Hexagonal Geometry Calculation*. ANL-95/40. Argonne National Laboratory (ANL). Argonne, Illinois, USA (pp. 66, 77).

Paradiseo team (2014). *Paradiseo: A Software Framework for Metaheuristics*. French Institute for Research in Computer Science and Automation (INRIA). URL: `paradiseo.gforge.inria.fr/` (p. 181).

Parejo, J., A. Ruiz-Cortés, S. Lozano, and P. Fernandez (2011). "Metaheuristic Optimization Frameworks: A Survey and Benchmarking". In: *Soft Computing—A Fusion of Foundations, Methodologies and Applications*. 10.1007/s00500-011-0754-8, pp. 1–35 (p. 180).

Park, T. K., H. G. Joo, C. H. Kim, and H. C. Lee (2009). "Multiobjective Loading Pattern Optimization by Simulated Annealing Employing Discontinuous Penalty Function and Screening Technique". In: *Nuclear Science and Engineering* 162.2, pp. 134–147 (p. 209).

Park, Y.-S., J.-H. Kim, and Y.-O. Lee (1994). "Establishing the Long-Term Fuel Management Scheme using Point Reactivity Model". In: *Journal of Nuclear Science and Technology* 31.10, pp. 1001–1010 (p. 19).

Parks, G. T. (1995). "Multiobjective PWR Reload Core Optimization using Genetic Algorithms". In: *Proceedings of the International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses*, pp. 615–624 (pp. 101, 201).

Parks, G. T. (1996). "Multiobjective Pressurized Water Reactor Reload Core Design by Nondominated Genetic Algorithm Search". In: *Nuclear Science and Engineering* 124.1, pp. 178–187 (pp. 101, 111, 202).

Parks, G. T. (1997). "Multiobjective Pressurised Water Reactor Reload Core Design using a Genetic Algorithm". In: *Artificial Neural Nets and Genetic Algorithms*. Ed. by G. D. Smith, N. C. Steele, and R. F. Albrecht. Norwich, UK: Springer-Verlag, pp. 53–57 (pp. 31, 101, 202).

Parks, G. T. (1987). "Optimisation of Advanced Gas-Cooled Reactor Fuel Performance by a Stochastic Method". In: *Nuclear Energy* 26, pp. 319–327 (pp. 31, 198).

Parks, G. T. (1990a). "Advances in Optimization and Their Applicability to Problems in the Field of Nuclear Science and Technology". In: *Advances in Nuclear Science and Technology* 21, pp. 195–253 (p. 199).

Parks, G. T. (1990b). "An Intelligent Stochastic Optimization Routine for Nuclear Fuel Cycle Design". In: *Nuclear Technology* 89.2, pp. 233–246 (p. 199).

Parks, G. T. and M. P. Knight (1997). "A Comparison of PANTHER Loading Pattern Optimization Search Options". In: *Topical Meeting on Advances in Nuclear Fuel Management II*. American Nuclear Society. Myrtle Beach, South Carolina, USA (p. 202).

Parks, G. T. and J. D. Lewins (1992). "In-Core Fuel Management and Optimization—The State of The Art". In: *Nuclear Europe World Scan* 41.4, pp. 3–4 (p. 199).

Parks, G. T. and A. Suppapitnarm (1999). "Multiobjective Optimization of PWR Reload Core Designs using Simulated Annealing". In: *Mathematics & Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*. Vol. 2. Madrid, Spain, pp. 1435–1444 (pp. 31, 204).

Parks, G. T., P. J. Turinsky, and G. I. Maldonado (1990). "Solving the PWR Reload Core Optimization Problem". In: *Scientific Excellence in Scientific Supercomputing, The IBM 1990 Contest Prize Papers*. Park City, Utah, USA, p. 281 (p. 199).

Pazirandeh, A. and S. Tayefi (2012). "Optimizing the Fuel Management in a VVER-1000 Reactor using an Artificial Neural Network". In: *Annals of Nuclear Energy* 42, pp. 112–118 (pp. 29, 211).

Pelykh, S. N., M. V. Maksimov, and G. T. Parks (2013). "A Method for VVER-1000 Fuel Rearrangement Optimization Taking Into Account Both Fuel Cladding Durability and Burnup". In: *Nuclear Engineering and Design* 257, pp. 53–60 (p. 212).

Pereira, C. M. N. A. and W. F. Sacco (2008). "A Parallel Genetic Algorithm with Niching Technique applied to a Nuclear Reactor Core Design Optimization Problem". In: *Progress in Nuclear Energy* 50.7. International Nuclear Atlantic Conference 2005, pp. 740–746 (p. 31).

Petrović, B. G. and S. H. Levine (1992). "Multicycle PWR In-Core Fuel Management Through One-and-a-half-Dimensional Core Modeling". In: *Transactions of the American Nuclear Society* 65, pp. 431–433 (p. 199).

Poon, P. W. and J. N. Carter (1995). "Genetic Algorithm Crossover Operators for Ordering Applications". In: *Computers & Operations Research* 22.1. Genetic Algorithms, pp. 135–147 (pp. 103, 111).

Poon, P. W. and G. T. Parks (1992). "Optimising PWR Reload Core Designs". In: *Parallel Problem Solving from Nature 2*. Elsevier Science Inc, pp. 371–380 (pp. 101, 111, 112, 114, 115, 131, 199).

Poon, P. W. and G. T. Parks (1993). "Application of Genetic Algorithms to In-Core Nuclear Fuel Management Optimization". In: *The Joint International Conference on Mathematical Methods and Super-Computing in Nuclear Applications*. Vol. 1. Karlsruhe, Germany, pp. 777–786 (pp. 31, 100, 101, 103, 109, 111, 200).

Poon, P. W. (1992). "The Genetic Algorithm Applied to PWR Reload Core Design". PhD Dissertation. UK: Cambridge University, Department of Engineering (p. 199).

Poursalehi, N., A. Zolfaghari, and A. Minuchehr (2013a). "Differential Harmony Search Algorithm to Optimize PWRs Loading Pattern". In: *Nuclear Engineering and Design* 257, pp. 161–174 (p. 212).

Poursalehi, N., A. Zolfaghari, and A. Minuchehr (2013b). "Multi-Objective Loading Pattern Enhancement of PWR Based on the Discrete Firefly Algorithm". In: *Annals of Nuclear Energy* 57, pp. 151–163 (pp. 31, 212).

Poursalehi, N., A. Zolfaghari, and A. Minuchehr (2013c). "PWR Loading Pattern Optimization using Harmony Search Algorithm". In: *Annals of Nuclear Energy* 53, pp. 288–298 (pp. 32, 212).

Poursalehi, N., A. Zolfaghari, A. Minuchehr, and H. K. Moghaddam (2013). "Continuous Firefly Algorithm Applied to PWR Core Pattern Enhancement". In: *Nuclear Engineering and Design* 258, pp. 107–115 (pp. 31, 212).

Poursalehi, N., A. Zolfaghari, A. Minuchehr, and K. Valavi (2013). "Self-Adaptive Global Best Harmony Search Algorithm Applied to Reactor Core Fuel Management Optimization". In: *Annals of Nuclear Energy* 62, pp. 86–102 (p. 212).

Python Software Foundation (2014). *Python*. Python Software Foundation. URL: `www.python.org` (p. 176).

Qt Project (2014). URL: `qt-project.org/` (p. 183).

Quist, A. J., E. de Klerk, C. Roos, and T. Terlaky (1998). *Finding Optimal Nuclear Reactor Core Reload Patterns using Nonlinear Optimization and Search Heuristics* (p. 203).

Quist, A. J., R. van Geemert, J. E. Hoogenboom, E. De Klerk, C. Roos, and T. Terlaky (1999). *Nuclear Reactor Fuel Management Optimization* (p. 204).

Quist, A. J., R. van Geemert, J. E. Hoogenboom, T. Illés, C. Roos, and T. Terlaky (1999). "Application of Nonlinear Optimization to Reactor Core Fuel Reloading". In: *Annals of Nuclear Energy* 26.5, pp. 423–448 (pp. 28, 204).

Quist, A. J., R. van Geemert, J. E. Hoogenboom, T. Illés, E. de Klerk, C. Roos, and T. Terlaky (1999). "Finding Optimal Nuclear Reactor Core Reload Patterns using Nonlinear Optimization and Search Heuristics". In: *Engineering Optimization* 32.2, pp. 143–176 (pp. 28, 204).

Quist, A. J., K. Roos, T. Terlaky, R. van Geemert, and E. Hoogenboom (2001). "Reloading Nuclear Reactor Fuel using Mixed-Integer Nonlinear Optimization". In: *Optimization and Engineering* 2.3, pp. 251–276 (pp. 28, 205).

Rahmani, Y., A. Pazirandeh, M. B. Ghofrani, and M. Sadighi (2012). "Calculation of the Deterministic Optimum Loading Pattern of the BUSHEHR VVER-1000 Reactor using the Weighting Factor Method". In: *Annals of Nuclear Energy* 49, pp. 170–181 (p. 211).

Rechenberg, I. (1965). *Cybernetic Solution Path of an Experimental Problem*. Technical Report 1122. Royal Aircraft Establishment (p. 31).

Reeves, C. R. (1993). *Modern Heuristic Techniques for Combinatorial Problems*. John Wiley & Sons, Inc. (p. 30).

Reuss, P. (2008). *Neutron Physics*. France: EDP Sciences (pp. 6, 36, 39, 42).

Rieck, T. A. (1974). "The Effect of Refueling Decisions and Engineering Constraints on the Fuel Management for a Pressurized Water Reactor". PhD Dissertation. Massachusetts: Massachusetts Institute of Technology, Department of Nuclear Engineering (p. 196).

Rieck, T. A., M. Benedict, and E. A. Mason (1974). *The Effect of Refueling Decisions and Engineering Constraints on the Fuel Management for a Pressurized Water Reactor*. Report MITNE-158. Massachusetts: Massachusetts Institute of Technology, Department of Nuclear Engineering (p. 196).

Rimpault, G., D. Plisson, J. Tommasi, R. Jacqmin, J.-M. Rieunier, D. Verrier, and D. Biron (2002). "The ERANOS Code and Data System for Fast Reactor Neutronic Analysis". In: *Proceedings of the International Conference on the New Frontiers of Nuclear Technology: Reactor Physics, Safety and High-Performance Computing (PHYSOR 2002)*. Seoul, Korea: Korean Nuclear Society (KNS) (pp. 71, 81).

Riverbank Computing Limited (2014). *What is PyQt?* Riverbank Computing Limited. URL: `www.riverbankcomputing.com/software/pyqt/intro` (p. 182).

Romanello, V., M. Salvatores, A. Schwenk-Ferrero, F. Gabrielli, W. Maschek, and B. Vezzoni (2011). "Comparative Study of Fast Critical Burner Reactors and Subcritical Accelerator Driven Systems and the Impact on Transuranics Inventory in a Regional Fuel Cycle". In: *Nuclear Engineering and Design* 241.1, pp. 433–443 (p. 6).

Ronen, Y., ed. (1986a). *CRC Handbook of Nuclear Reactors Calculations*. Vol. I. Boca Raton, Florida, USA: CRC Press, Inc. (p. 41).

Ronen, Y., ed. (1986b). *CRC Handbook of Nuclear Reactors Calculations*. Vol. II. Boca Raton, Florida, USA: CRC Press, Inc. (p. 41).

Ronen, Y., ed. (1986c). *CRC Handbook of Nuclear Reactors Calculations*. Vol. III. Boca Raton, Florida, USA: CRC Press, Inc. (p. 41).

Rosset, F. D. and J. C. Barral (1993). "Fuel Management and Reloading Optimization at EdF". In: *Transactions of the American Nuclear Society* 69, pp. 418–419 (p. 200).

Rothleder, B. M., G. R. Poetschat, W. S. Faught, and V. J. Eich (1988). "The Potential for Expert System Support in Solving the Pressurized Water Reactor Fuel Shuffling Problem". In: *Nuclear Science and Engineering* 100.4, pp. 440–450 (pp. 30, 198).

Sacco, W. F., N. Henderson, A. C. Rios-Coelho, M. M. Ali, and C. M. N. A. Pereira (2009). "Differential Evolution Algorithms Applied to Nuclear Reactor Core Design". In: *Annals of Nuclear Energy* 36.8, pp. 1093–1099 (p. 209).

Sackett, J. (2012). "Fast Spectrum Reactors". In: ed. by A. E. Waltar, D. R. Todd, and P. V. Tsvetkov. 2nd ed. Springer. Chap. General Safety Considerations, pp. 411–426 (p. 13).

Sadighi, M., S. Setayeshi, and A. A. Salehi (2002a). "Neutron Flux Flattening in PWRs using Neural Networks in Fuel Management". In: *IEEE Transactions on Nuclear Science* 49, pp. 1574–1578 (p. 206).

Sadighi, M., S. Setayeshi, and A. A. Salehi (2002b). "PWR Fuel Management Optimization using Neural Networks". In: *Annals of Nuclear Energy* 29.1, pp. 41–51 (pp. 29, 206).

Safarzadeh, O., A. Zolfaghari, A. Norouzi, and H. Minuchehr (2011). "Loading Pattern Optimization of PWR Reactors using Artificial Bee Colony". In: *Annals of Nuclear Energy* 38.10, pp. 2218–2226 (p. 211).

Safarzadeh, O., A. Zolfaghari, M. Zangian, and O. Noori-kalkhoran (2014). "Pattern Optimization of PWR Reactor using Hybrid Parallel Artificial Bee Colony". In: *Annals of Nuclear Energy* 63, pp. 295–301 (pp. 31, 213).

Salvatores, M. (1986). "CRC Handbook of Nuclear Reactors Calculations". In: ed. by Y. Ronen. Vol. III. Boca Raton, Florida, USA: CRC Press, Inc. Chap. Fast Reactor Calculations, pp. 263–364 (pp. 42, 78).

Santamarina, A., D. Bernard, P. Blaise, M. Coste, A. Courcelle, T. D. Huynh, C. Jouanne, C. Leconte, O. Litaize, S. Mengelle, G. Noguère, J.-M. Ruggiéri, O. Sérot, J. Tommasi, C. Vaglio, and J.-F. Vidal (2009). *The JEFF-3.1.1 Nuclear Data Library*. Report NEA No. 6807. Paris, France: Nuclear Energy Agency (OECD-NEA) (pp. 84, 225).

Sarker, A. R. and S. C. Newton (2008). *Optimization Modelling: A Practical Approach*. Florida, USA: CRC Press, Inc. (pp. 24, 25).

Sarotto, M., R. Fernandez, E. Malambu, A. Stankovskiy, E. Bubelis, R. Dagan, W. Jaeger, A. Travleev, M. Becker, F. Martín-Fuertes, M. Vázquez, F. Álvarez Velarde, A. Ferrari, S. DiMaria, L. Sabathé, and M. Ottolini (2012). *FASTEF Design Changes to Operate in Critical Mode*. Technical Report UTFISSM-P9P0-020. Confidential. ENEA Centro Ricerche Bologna (pp. 45, 54, 55, 57, 82, 91, 93, 95).

Sarotto, M., S. Vanmaercke, F. Martín-Fuertes, M. Vázquez, F. Álvarez, E. González, C. Artioli, G. Grasso, and L. Sabathé (2010). *D1.2 Analysis of MYRRHA-FASTEF to Operate in a Critical Mode*. Technical Report UTFISSM-P9P0-001. Confidential. ENEA Centro Ricerche Bologna (p. 51).

Sarotto, M. (2011). *Critical Core Design of MYRRHA-FASTEF: Refinement of Shutdown Systems and Neutronic Characterisation of the Equilibrium Sub-Cycle Core*. Technical Report UTFISSM-P9P0-011. Confidential. ENEA Centro Ricerche Bologna (pp. 68, 69, 76, 81–84, 89–95, 219–221, 223).

Sarotto, M., D. Castelliti, R. Fernandez, D. Lamberts, M. Edouard, A. Stankovskiy, W. Jaeger, M. Ottolini, F. Martin-Fuertes, L. Sabathé, L. Mansani, and P. Baeten (2013). "The MYRRHA-FASTEF Cores Design for Critical and Sub-Critical Operational Modes (EU FP7 Central Design Team Project)". In: *Nuclear Engineering and Design* 265, pp. 184–200 (pp. 44, 45, 48, 51, 52, 56).

Sato, D. (2005). "Development of PWR Fuel Loading Pattern Search Tool (Pearls^{TM})". In: *Mitsubishi Heavy Industries Technical Review* 42 (p. 207).

Sauar, T. O. (1971). "Application of Linear Programming to In-Core Fuel Management Optimization in Light Water Reactors". In: *Nuclear Science and Engineering* 46.2, pp. 274–283 (pp. 28, 196).

Sauer, I. L. and M. J. Driscoll (1985). *A Core Reload Pattern and Composition Optimization Methodology for Pressurized Water Reactors*. Report MITNE-266. Massachusetts, USA: Massachusetts Institute of Technology, Department of Nuclear Engineering (p. 198).

Schirru, R., A. M. M. de Lima, and M. D. Machado (2006). "Parallel Evolutionary Methods Applied to a PWR Core Reload Pattern Optimization". In: *Proceedings of the Seventh International Conference on Applied Artificial Intelligence*. FLINS. Italy (pp. 184, 208).

Schirru, R., C. M. N. A. Pereira, and F. C. Chapot J. L. C. da Silva (1997). "A Genetic Solution for Combinatorial Problems—The Nuclear Core Reload Example". In: *Proceedings of The 11th Brazilian Meeting on Reactor Physics and Thermal Hydraulics*, pp. 357–360 (p. 202).

Schlünz, E. B., R. H. Prinsloo, and P. M. Bokov (2013). "In-Core Fuel Management Optimisation within the OSCAR-4 Code System". In: *Proceeding of the European Research Reactor Conference (RRFM-2013)*, pp. 381–390 (pp. 27, 32, 213).

Sekimizu, K. (1975). "Optimization of In-Core Fuel Management and Control Rod Strategy in Equilibrium Fuel Cycle". In: *Journal of Nuclear Science and Technology* 12.5, pp. 287–296 (p. 197).

Shatilla, Y. A., D. C. Little, J. A. Penkrot, and R. A. Holland (2000). "In-Core Fuel Management with Biased Multiobjective Function Optimization". In: *Nuclear Technology* 130.3, pp. 282–295 (pp. 174, 204).

Shaukat, N., S.-I. Ahmad, A. Majeed, N. Ahmad, and B. Mohsin (2010). "Optimization of Core Reload Pattern for PARR-1 using Evolutionary Techniques". In: *Nuclear Engineering and Design* 240.10, pp. 2831–2835 (pp. 27, 210).

Shvedov, M. O. and L. A. Goncharov (1992). "Method and Code for Fuel Reloading Optimization in Transfer Regime of Fast Reactor Operation". In: *Proceedings of the 1992 Topical Meeting on Advances in Reactor Physics*. Vol. 23. Westinghouse Savannah River Co., Aiken, South California, USA: American Nuclear Society (ANS), pp. 178–186 (pp. 27, 199).

Siegelmann, H. T., E. Nissan, and A. Galperin (1997). "A Novel Neural/Symbolic Hybrid Approach to Heuristically Optimized Fuel Allocation and Automated Revision of Heuristics in Nuclear Engineering". In: *Advances in Engineering Software* 28.9, pp. 581–592 (p. 202).

Silva, M. H. da and R. Schirru (2010). "Optimization of Nuclear Reactor Core Fuel Reload using the New Quantum PBIL". In: *Annals of Nuclear Energy* 38.2–3, pp. 610–614 (pp. 31, 210).

Silva, M. H. da and R. Schirru (2014). "A Self-Adaptive Quantum PBIL Method for the Nuclear Reload Optimization". In: *Progress in Nuclear Energy* 74, pp. 103–109 (pp. 31, 213).

Silva, M. H. da, R. Schirru, and A. M. M. de Lima (2011). "QACO_Alpha Applied to the Nuclear Reactor Core Fuel Reload Optimization". In: *Progress in Nuclear Energy* 53.1, pp. 80–85 (pp. 31, 121, 211).

Šmuc, T., B. G. Petrović, and N. Urli (1993). "Automation of the Loading Pattern Search Process". In: *Nuclear Energy in Central Europe: Present and Perspectives*. Portorož, Slovenia, pp. 98–103 (p. 200).

Šmuc, T., D. Pevec, and B. G. Petrović (1994). "Annealing Strategies for Loading Pattern Optimization". In: *Annals of Nuclear Energy* 21.4, pp. 325–336 (p. 201).

Spencer, H. (1864). *The Principle of Biology*. Williams and Norgate (p. 109).

Stankovskiy, A. (2013). *Private Communication* (pp. 78, 225, 226).

Stankovskiy, A. and G. Van den Eynde (2011). *ALEPH 2.1 – A Monte Carlo Burn-Up Code. User's Manual*. SCK•CEN-R-5267. Technical Report. SCK•CEN. Mol, Belgium (pp. 7, 81).

Stankovskiy, A., G. Van den Eynde, and T. Vidmar (2011). "Development and Validation of ALEPH Monte Carlo Burn-Up Code". In: *Nucler Measurements, Evaluations and Applications—NEMEA-6—Workshop Proceedings*. NEA/NSC/DOC(2011)4. OECD. Krakow, Poland, pp. 161–169 (p. 7).

Stevens, J. G., K. S. Smith, and T. J. Downar (1991). "An Exhaustive Data Base for Comparing PWR Core Design. Optimization Algorithms". In: *Transactions of the American Nuclear Society* 64, p. 527 (p. 199).

Stevens, J. G., K. S. Smith, K. R. Rempe, and T. J. Downar (1993). "Simplified Automated Loading Pattern Generation for Multicycle Analysis". In: *Transactions of the American Nuclear Society* 69, pp. 420–422 (p. 200).

Stevens, J. G., K. S. Smith, K. R. Rempe, and T. J. Downar (1995). "Optimization of Pressurized Water Reactor Shuffling by Simulated Annealing with Heuristics". In: *Nuclear Science and Engineering* 121.1, pp. 67–88 (pp. 31, 201).

Stevens, J. G. and K. R. Rempe (2003). "Equilibrium Cycle Analysis with XIMAGE/SIMAN". In: *Proceedings of the Advances in Nuclear Fuel Management III (ANFM 2003)*. Hilton Head Island, South Carolina, USA: American Nuclear Society (ANS) (pp. 174, 206).

Stevens, J. G. (1995). "A Hybrid Method for In-Core Optimization of Pressurized Water Reactor Reload Core Design". PhD Dissertation. Lafayette, Indiana, USA: Purdue University (p. 201).

Stichelbaut, F. and Y. Jongen (2011). "Design of Accelerator-Based Solutions to produce ^{99}Mo Using Lowly-Enriched Uranium". In: *Nuclear Science and Technology* 2, pp. 284–288 (p. 44).

Stillman, J. A., Y. A. Chao, and T. J. Downar (1989). "The Optimum Fuel and Power Distribution for a Pressurized Water Reactor Burnup Cycle". In: *Nuclear Science and Engineering* 103.4, pp. 321–333 (p. 199).

Story, G. and R. Grow (1985). "A Microcomputer Workstation for Design Analysis of PWR Fuel Loading Patterns". In: *Transaction of the American Nuclear Society*. Vol. 50, pp. 97–98 (p. 198).

Stout, R. B. and A. H. Robinson (1973). "Determination of Optimum Fuel Loading in Pressurized Water Reactors using Dynamic Programming". In: *Nuclear Technology* 20, p. 86 (p. 196).

Stover, R. L. and A. Sesonske (1969). "Optimization of BWR Fuel Management using an Accelerated Exhaustive Search Algorithm". In: *Journal of Nuclear Energy* 23.11-12, pp. 673–682 (p. 196).

Studsvik (2014). *XIMAGE*. Studsvik Scandpower, Inc. URL: `http://www.studsvik.com/en/Business-Areas/Operating-Efficiency/Nuclear-Fuel-Analysis-Software/Loading-Pattern-Design/XIMAGE/` (p. 174).

Suh, J. S. (1989). "Optimized Automatic Reload Programs for PWR's using the Haling Power Distribution". PhD Dissertaion. Raleigh, Pennsylvania, USA: Pennsylvania State University (p. 199).

Suh, J. S. and S. H. Levine (1990). "Optimized Automatic Reload Program for Pressurized Water Reactors using Simple Direct Optimization Techniques". In: *Nuclear Science and Engineering* 105.4, pp. 371–382 (p. 199).

Suzuki, A. and R. Kiyose (1971a). "Application of Linear Programming to Refuelling Optimization for Light Water Reactors". In: *Nuclear Science and Engineering* 46.1, pp. 112–130 (pp. 28, 196).

Suzuki, A. and R. Kiyose (1971b). "Maximizing the Average Fuel Burn-Up over Entire Core: A Poison Management Optimization Problem for Multizone Light Water Reactor Cores". In: *Nuclear Science and Engineering* 44, pp. 121–134 (p. 196).

Tahara, Y., K. Hamamoto, M. Takase, and K. Suzuki (1991). "Computer Aided System for Generating Fuel Shuffling Configurations Based on Knowledge Engineering". In: *Journal of Nuclear Science and Technology* 28.5, pp. 399–408 (pp. 30, 199).

Talbi, E.-G. (2009). *Metaheuristics: From Design to Implementation*. Wiley (pp. 30, 31, 99, 100, 103, 105, 109, 111, 113–117, 127, 130, 180, 181).

Tang, Y. S., R. D. Coffield Jr., and R. A. Markley (1978). *Thermal Analysis of Liquid Metal Fast Breeder Reactors*. American Nuclear Society (ANS) (p. 34).

Tanker, E. and A. Tanker (1994). "Application of Genetic Algorithm to Core Reload Pattern Optimization". In: *Proceedings of the International Conference on Reactor Physics and Reactor Computations*, p. 559 (pp. 111, 201).

Tayefi, S. and A. Pazirandeh (2013). "Evaluation of Applying Axial Variation of Enrichment Distribution Method and Radial Variation of Enrichment Distribution in VVER-1000 Reactor using a Hopfield Neural Network to Optimize Fuel Management". In: *Progress in Nuclear Energy* 64, pp. 47–53 (p. 213).

Terney, W. B. and E. A. Williamson Jr. (1982). "The Design of Reload Cores using Optimal Control Theory". In: *Nuclear Science and Engineering* 82, pp. 260–288 (p. 197).

Thakur, A., B. Singh, and P. D. Krishnani (2013). "In-Core Fuel Management for AHWR". In: *Annals of Nuclear Energy* 57, pp. 47–58 (p. 213).

Todreas, N. E. and M. S. Kazimi (1990a). *Nuclear Systems I: Thermal Hydraulic Fundamentals*. Vol. I. Taylor&Francis (pp. 34, 51).

Todreas, N. E. and M. S. Kazimi (1990b). *Nuclear Systems II: Elements of Thermal Hydraulic Design*. Vol. II. Taylor&Francis (p. 34).

Toppel, B. J. (1990). *A User's Guide for the REBUS-3 Fuel Cycle Analyses Capability*. ANL-83-2. Argonne National Laboratory (ANL). Argonne, Illinois, USA (p. 67).

Toshinsky, V. G., H. Sekimoto, and G. I. Toshinsky (1999). "Multiobjective Fuel Management Optimization for Self-Fuel-Providing LMFBR using Genetic Algorithms". In: *Annals of Nuclear Energy* 26.9, pp. 783–802 (pp. 78, 101, 111, 114, 204).

Tsvetkov, P., A. Waltar, and M. Salvatores (2012). "Fast Spectrum Reactors". In: ed. by A. E. Waltar, D. R. Todd, and P. V. Tsvetkov. 2nd ed. Springer. Chap. Fuel Management, pp. 135–185 (pp. 6, 12, 17, 78).

Turinsky, P. J. and G. H. Hobson (1986). "Automatic Determination of Pressurized Water Reactor Core Loading Patterns that Maximize Beginning-of-Cycle Reactivity within Power-Peaking and Burnup Constraints". In: *Nuclear Technology* 74, p. 5 (p. 198).

Turinsky, P. J. and G. T. Parks (1999). "Advances in Nuclear Fuel Management for Light Water Reactors". In: *Advances in Nuclear Science and Technology* 26, pp. 137–165 (p. 204).

Turinsky, P. J. (1999). "Mathematical Optimization for Incore Fuel Management Decisions: Status and Trends". In: *Atw-International Zeichrift für Kernenergie* 44.7, pp. 414–415, 454–458 (pp. 32, 204).

Turinsky, P. J. (2005). "Nuclear Fuel Management Optimization: A Work in Progress". In: *Nuclear Technology* 151.1, pp. 3–8 (pp. 27, 207).

Turinsky, P. J. (2010). "Handbook of Nuclear Engineering". In: ed. by D. G. Cacuci. Springer. Chap. Core Isotopic Depletion and Fuel Management, pp. 1241–1312 (pp. 16, 17, 21, 210).

Turinsky, P. J., P. M. Keller, and H. S. Abdel-Khalik (2005). "Evolution of Nuclear Fuel Management and Reactor Operational Aid Tools". In: *Nuclear Engineering and Technology* 37.1, pp. 79–90 (p. 207).

Van de Velde, A., J. K. Axmann, and H.-W. Bolloni (1999). "Recent Issues in Automated Fuel Management Optimization with Adaptive Evolutionary Algorithms". In: *Mathematical Methods and Supercomputing in Nuclear Applications*. Madrid (p. 204).

Van den Eynde, G., K. Nishihara, E. Malambu, R. Dagan, M. Schikorr, D. Struwe, G. Rimpault, F. Alvarez, and M.-C. Vicente-Vicente (2009). *Core Neutronics Including Calculation of Safety Related Parameters of XT-ADS—Core Cycle Analysis of XT-ADS*. Technical Report ANS/RMS/GVdE/gvde.3910.A0B044000-0123 825/09-08. Confidential. Mol, Belgium: SCK•CEN (pp. 51, 56).

Van den Eynde, G., A. Stankovskiy, L. Fiorito, and M. Broustaut (2013). "Development and Validation of ALEPH-2 Monte Carlo Burnup Code". In: *Proceedings of the 2013 International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2013)*. SCK•CEN. Sun Valley, Idaho, USA: ANS, pp. 2566–2577 (p. 81).

Van den Eynde, G. (2003). *Calculating the MYRRHA Core using DIF3D 8.0*. Technical Report. Mol, Belgium: SCK•CEN (p. 62).

van Geemert, R. (1995). *Evaluation and Optimization of Fuel Assembly Shuffling Schemes for a Nuclear Reactor Core*. Technical Report IRI-131-95-016. Delft, Netherlands: TU Delft, Interfacultair Reactor Instituut (p. 201).

van Geemert, R., Q. A. J., and J. E. Hoogenboom (1997). "Fuel Shuffling Optimization for the Delft Research Reactor". In: *Proceeding of the 1st International Topical Meeting on Research Reactor Fuel Management (RRFM '97)*. European Nuclear Society (ENS). Bruges, Belgium, pp. 65–69 (p. 202).

van Geemert, R., A. J. Quist, and E. J. Hoogenboom (1996). "Reload Pattern Optimization by Application of Multiple Cyclic Interchange Algorithms". In: *PHYSOR'96*. Mito, Japan (p. 201).

van Geemert, R., A. J. Quist, J. E. Hoogenboom, and H. P. M. Gibcus (1998). "Research Reactor In-Core Fuel Management Optimization by Application of Multiple Cyclic

Interchange Algorithms". In: *Nuclear Engineering and Design* 186.3, pp. 369–377 (pp. 27, 203).

van Geemert, R. (1999). *Reload Pattern Optimization by Application of Heuristic Search and Perturbation Theoretical Methods*. Delft University Press (p. 203).

Verhagen, F. C. M., M. Van der Schaar, W. J. M. de Kruijf, T. F. H. Van de Wetering, and R. D. Jones (1997). "ROSA, a Utility Tool for Loading Pattern Optimization". In: *Advances in Nuclear Fuel Management II*. American Nuclear Society. Myrtle Beach, South Carolina, USA (p. 202).

Verhagen, F. C. M. and M. Van der Schar (1993). "Simulated Annealing in LWR Fuel Management". In: *TOPNUX'93*. Vol. 2, pp. 37–39 (p. 200).

Verhagen, F. C. M. and P. H. Wakker (2003). "ROSA, a Flexible Loading Pattern Optimization Tool for PWRs". In: *Advances in Nuclear Fuel Management III*. American Nuclear Society. Hilton Head Island, South Carolina, USA (pp. 174, 206).

Verhagen, F. C. M., P. H. Wakker, and J. T. van Bloois (2003). "Minimizing PWR Reloading Time by Optimizing Core Design and Reshuffling Sequence". In: *Proceedings of the TopFuel 2003 Conference*. Würzburg, Germany (p. 206).

W3C (2014). *Extensible Markup Language (XML)*. World Wide Web Consortium (W3C). URL: `www.w3.org/XML/` (p. 182).

Wade, D. C. and W. B. Terney (1971). "Optimal Control of Nuclear Reactor Depletion". In: *Nuclear Science and Engineering* 45.2, pp. 199–217 (p. 196).

Waintraub, M., R. Schirru, and C. M. N. A. Pereira (2009). "Multiprocessor Modeling of Parallel Particle Swarm Optimization applied to Nuclear Engineering Problems". In: *Progress in Nuclear Energy* 51.6-7, pp. 680–688 (pp. 184, 209).

Wall, I. and H. Fenech (1965). "The Application of Dynamic Programming to Fuel Management Optimization". In: *Nuclear Science and Engineering* 22, pp. 285–297 (p. 196).

Waltar, A. E., D. R. Todd, and P. V. Tsvetkov, eds. (2012). *Fast Spectrum Reactors*. 2nd ed. Springer (p. 66).

Wang, C.-D. and C. Lin (2009). "Automatic Boiling Water Reactor Loading Pattern Design using Ant Colony Optimization Algorithm". In: *Annals of Nuclear Energy* 36.8, pp. 1151–1158 (pp. 31, 121, 124, 209).

Wantz, O. (2000). *Combination of Synchronous and Asynchronous Evolutionary Algorithm for Loading Pattern Optimization*. Technical Report A1C-1308507-0. Confidential. Erlangen, Germany: Siemens (p. 204).

White, J. R., D. M. Chapman, and D. Biswas (1986). "Fuel Management Optimization Based on Generalized Perturbation Theory". In: *Proceeding of the Topical Meeting on Advances in Fuel Management*. Pinehurst, North Carolina, USA: American Nuclear Society (ANS), pp. 525–532 (p. 198).

X-5 Monte Carlo Team (2003). *MCNP—A General Monte Carlo N-Particle Transport Code, Version 5, Volume II: User's Guide*. LA-UR-03-1987. Los Alamos National Laboratory. Los Alamos, New Mexico, USA (p. 7).

X-5 Monte Carlo Team (2008). *MCNPX User's Manual: Version 2.6.0*. LA-CP-07-1473. Los Alamos National Laboratory. Los Alamos, New Mexico, USA (pp. 7, 81).

X-5 Monte Carlo Team (2011). *MCNPX-2.7.E Extensions*. LA-UR-11-1502. Los Alamos National Laboratory. Los Alamos, New Mexico, USA (p. 81).

Yadav, R. D. S. and H. P. Gupta (2012). "Optimization Studies of Fuel Loading Pattern for a Typical Pressurized Water Reactor (PWR) using Particle Swarm Method". In: *Annals of Nuclear Energy* 50, pp. 117–125 (p. 212).

Yamamoto, A. (1996). "Loading Pattern Optimizations using Genetic Algorithms". In: *PHYSOR'96*. Vol. 3. Mito, Japan, pp. 48–56 (p. 202).

Yamamoto, A. (1997). "A Quantitative Comparison of Loading Pattern Optimization Methods for In-Core Fuel Management of PWR". In: *Nuclear Science and Technology* 34.4, pp. 339–347 (pp. 30, 101, 202).

Yamamoto, A. (1998). "Study on Advanced In-Core Fuel Management for Pressurized Water Reactors using Loading Pattern Optimization Methods". PhD Dissertation. Kyoto, Japan: Kyoto University (p. 203).

Yamamoto, A. (2001a). "Recent Activities of Loading Pattern Optimization Research in Japan". In: *Transactions of the American Nuclear Society* 84 (p. 205).

Yamamoto, A. (2001b). "Recent Enhancements of the INSIGHT Integrated In-Core Fuel Management Tool". In: *Proceedings of the Korean Nuclear Society spring meeting*. Korean Nuclear Society (KAERI). Taejon, Republic of Korea (p. 205).

Yamamoto, A. and H. Hashimoto (2000). "Application of Temperature Parallel Simulated Annealing to Loading Pattern Optimizations of Pressurized Water Reactors". In: *Nuclear Science and Engineering* 136.2, pp. 247–257 (pp. 31, 205).

Yamamoto, A. and H. Hashimoto (2002). "Application of the Distributed Genetic Algorithm for In-Core Fuel Optimization Problems under Parallel Computational Environment". In: *Journal of Nuclear Science and Technology* 39.12, pp. 1281–1288 (pp. 31, 184, 206).

Yamamoto, A. and K. Kanda (1997). "Comparison Between Equilibrium Cycle and Successive Multicycle Optimization Methods for In-Core Fuel Management of Pressurized Water Reactors". In: *Journal of Nuclear Science and Technology* 34.9, pp. 882–892 (p. 203).

Yamamoto, A., H. Noda, N. Ito, and T. Maruyama (1997). "INSIGHT: An Integrated Scoping Analysis Tool for In-core Fuel Management of PWR". In: *Journal of Nuclear Science and Technology* 34.8, pp. 847–855 (pp. 101, 203).

Yamamoto, A., E. Sugimura, Y. Kitamura, and Y. Yamane (2004). "Simultaneous Loading Patterns Optimization for Two Successive Cycles of Pressurized Water Reactors". In: *Journal of Nuclear Science and Technology* 41.11, pp. 1065–1074 (p. 207).

Yang, W. S. and T. A. Taiwo (2004). "Status of Reactor Analysis Methods and Codes in the U.S.A". In: *Proceedings of the International Conference on The Physics of Fuel Cycles and Advanced Nuclear Systems: Global Developments (PHYSOR 2004)*. Chicago, Illinois, USA: American Nuclear Society (ANS) (p. 64).

Yang, W. S. (2012). "Fast Reactor Physics and Computational Methods". In: *Nuclear Engineering and Technology* 44.2, pp. 177–198 (pp. 65–67).

Yang, W. S. and H. S. Khalil (2001). "Blanket Design Studies of a Lead-Bismuth Eutectic-Cooled Accelerator Transmutation of Waste System". In: *Nuclear Technology* 135, pp. 162–182 (p. 85).

Yeh, S. H., J. Weisman, and A. Y. Ying (1983). "An Automated Procedure for Selecting Boiling Water Reactor Refueling Policies Following Operational Problems or Changes". In: *Nuclear Technology* 61.1, pp. 78–92 (p. 197).

Yilmaz, S., G. Kvaall, S. Sutton, and G. Huff (2011). "Advanced ePrometheus^{TM} GNF Fuel Cycle Optimization Tool". In: *Progress in Nuclear Energy* 53.6, pp. 562–565 (p. 211).

Zavaljsevski, N. (1990). "A Model for Fuel Shuffling and Burnable Absorbers Optimization in Low Leakage PWR's". In: *Annals of Nuclear Energy* 17.4, pp. 217–220 (p. 199).

Zhao, J., B. Knight, E. Nissan, and A. Soper (1998). "FuelGen: A Genetic Algorithm Based System for Fuel Loading Pattern Design in Nuclear Power Reactors". In: *Expert Systems with Applications* 14, pp. 461–470 (p. 203).

Zhian, L. and S. H. Levine (1993). "AUTOLOAD, an Expert System Code that Automatically and Optimally Reloads a PWR". In: *Transactions of the American Nuclear Society* 69, pp. 456–458 (p. 200).

Zimmerman, S. G. and J. A. Umbarger (1997). "Design of Aggresive Loading Patterns with X-IMAGE/SIMAN for Palo Verde Nuclear Generation Station". In: *Proceedings of the Advances in Nuclear Fuel Management II (ANFM-II)*. Vol. 2. Myrtle Beach, South Carolina, USA: American Nuclear Society, pp. 16–17 (p. 203).

Ziver, A. K., J. N. Carter, C. C. Pain, C. R. E. de Oliveira, and A. J. H. Goddard (2001a). *The Program GAOPT Version 3*. Report. London, UK: Imperial College London, Applied Modelling and Computation Group (p. 205).

Ziver, A. K., J. N. Carter, C. C. Pain, C. R. E. de Oliveira, and A. J. H. Goddard (2001b). *The Program GAOPT Version 3, a User's Manual for AGR Refuelling Optimiser*. Imperial College London, Applied Modelling and Computation Group. London, UK (p. 205).

Ziver, A. K., J. N. Carter, C. C. Pain, C. R. E. de Oliveira, A. J. H. Goddard, and R. S. Overton (2002). "On the Use of Artificial Neural Networks in Loading Pattern Optimisation of Advanced Gas-Cooled Reactors". In: *International Conference on New Frontiers of Nuclear Technology*. Seoul, Korea (p. 206).

Ziver, A. K., C. C. Pain, J. N. Carter, C. R. E. de Oliveira, A. J. H. Goddard, and R. S. Overton (2004). "Genetic Algorithms and Artificial Neural Networks for Loading Pattern Optimisation of Advanced Gas-Cooled Reactors". In: *Annals of Nuclear Energy* 31.4, pp. 431–457 (pp. 27, 29, 31, 101, 108, 109, 111, 115, 207).

FACULTY OF ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE
NUMERICAL ANALYSIS AND APPLIED MATHEMATICS SECTION
Celestijnenlaan 200A box 2402
B-3001 Heverlee
david.jaluvka@gmail.com